

TAXONOMÍA SOBRE ATRIBUTOS DE CALIDAD EN APLICACIONES DISEÑADAS
PARA SER DESPLEGADAS EN LA NUBE

CARLOS ANDRÉS GUERRERO ALARCÓN

UNIVERSIDAD PONTIFICIA BOLIVARIANA
ESCUELA DE INGENIERÍAS
DOCTORADO EN INGENIERÍA
MEDELLÍN
2019

TAXONOMÍA SOBRE ATRIBUTOS DE CALIDAD EN APLICACIONES DISEÑADAS
PARA SER DESPLEGADAS EN LA NUBE

CARLOS ANDRÉS GUERRERO ALARCÓN

Trabajo de grado para optar por el título de Doctor en Ingeniería

Director de tesis de doctorado

PhD. JORGE MARIO LONDOÑO PELÁEZ

Doctor en Ciencias de la Computación

UNIVERSIDAD PONTIFICIA BOLIVARIANA

ESCUELA DE INGENIERÍAS

DOCTORADO EN INGENIERÍA

MEDELLÍN

2019

19 de noviembre de 2019

Carlos Andrés Guerrero Alarcón

“Declaro que esta tesis no ha sido presentada para optar a un título, ya sea en igual forma o con variaciones, en esta o cualquier otra universidad” Art 82 Régimen Discente de Formación Avanzada.

Firma

A handwritten signature in black ink, appearing to read 'Carlos', written in a cursive style. The signature is positioned above a horizontal line that spans the width of the page.

CONTENIDO

<u>1. INTRODUCCIÓN</u>	1
1.1. Planteamiento del problema	1
1.2. Objetivos de la investigación	4
1.2.1. Objetivo general	4
1.2.2. Objetivos específicos	4
1.3. Justificación	5
1.4. Método de investigación	6
1.5. Alcance de la tesis	11
1.6. Contribuciones de la tesis	12
1.7. Estructura del documento	13
<u>2. TRABAJOS RELACIONADOS</u>	14
2.1. Calidad en computación en la nube	14
2.1.1. Taxonomías	14
2.1.2. Modelos de calidad	16
2.2. Índice de medición del servicio	19
2.3. Normatividad relacionada con calidad de software	20
2.3.1. Organismos de normalización	20
2.3.2. Modelo de calidad seleccionado en este estudio.....	23
2.3.3. Norma ISO/IEC 25010	24
2.3.4. Norma ISO/IEC 25012	25
<u>3. VOCABULARIOS CONTROLADOS</u>	31
3.1. Concepto	31
3.2. Sistemas de organización del conocimiento	31
3.3. Principios de los vocabularios controlados	32
3.4. Garantía literaria	33
<u>4. PROTOCOLO PARA LA CONSTRUCCIÓN DEL VOCABULARIO</u>	34
4.1. Necesidad del protocolo	34
4.2. Planificación de la revisión sistemática	34
4.2.1. Necesidad de la revisión	34
4.2.2. Identificación de la investigación	34
4.3. Desarrollo del protocolo	35
4.3.1. Repositorios	35
4.3.2. Palabras claves.....	36
4.3.3. Ecuación de búsqueda.....	36

4.3.4. Ejecución de la ecuación de búsqueda	37
4.3.5. Preselección de trabajos	39
4.3.6. Evaluación de la calidad de los trabajos preseleccionados	43
4.3.7. Línea de base para la extracción	44
4.3.8. Extracción de datos.....	50
4.4. Resumen de la ejecución del protocolo	53
4.5. Amenazas a la validez del trabajo realizado.....	56

5. PRESENTACIÓN DE LA TAXONOMÍA..... 57

5.1. Principios.....	57
5.2. Tipo de relaciones utilizadas.....	57
5.2.1. Relación genérica	57
5.2.2. Relación todo - parte.....	58
5.3. Tipo de presentación	58
5.4. Una nueva taxonomía sobre atributos de calidad en la nube	59
5.4.1. Característica “1.1 Accountability”	60
5.4.2. Característica “1.2 Agility”	65
5.4.3. Característica “1.3 Assurance”	71
5.4.4. Característica “1.4 Empathy”	78
5.4.5. Característica “1.5 Financiamiento”	80
5.4.6. Característica “1.6 Performance”	82
5.4.7. Característica “1.7 Security and Privacy”	87
5.4.8. Característica “1.8 Usability”	94
5.4.9. Característica “1.9 User experience”	97

6. MODELO DE CALIDAD..... 98

6.1. Alcance	98
6.2. El nuevo modelo y sus vistas.....	98
6.2.1. Vista de usuario	101
6.2.2. Vista de proveedor	102
6.2.3. Vista desarrollador	103
6.3. Instrumento para la validación del modelo de calidad	104
6.3.1. Contexto industria de software	104
6.3.2. Contexto academia	105
6.4. Selección de la muestra.....	105
6.4.1. Contexto industria de software	105
6.4.2. Contexto academia	106
6.5. Aplicación del instrumento	107
6.5.1. Contexto industria de software	107
6.5.2. Contexto academia	108
6.6. Definición de elementos claves para el análisis de resultados	108
6.6.1. Análisis por realizar	108
6.6.2. Objetivo del análisis	108
6.6.3. Justificación	108
6.7. Resultados de la aplicación del instrumento	109

6.7.1. Contexto industria de software	109
6.7.2. Contexto academia	117
6.8. Guía para la selección de atributos de calidad	117
6.8.1. Vista desarrollador	118
6.8.2. Vista usuario	119
6.8.3. Vista proveedor	119
<u>7. CONCLUSIONES</u>	<u>120</u>
7.1. Conclusiones por objetivo	120
7.2. Conclusiones respecto a la hipótesis.....	122
7.3. Conclusiones generales	123
<u>8. TRABAJOS FUTUROS</u>	<u>125</u>
<u>9. REFERENCIAS.....</u>	<u>126</u>
<u>APÉNDICE A</u>	<u>149</u>
<u>APÉNDICE B</u>	<u>185</u>

LISTA DE FIGURAS

Figura 1. Espina-pescado planteamiento problema	3
Figura 2. Fases del método	7
Figura 3. Alcance de la tesis	12
Figura 4. Cuadrante mágico para Cloud Infrastructure as a Service, Worldwide.....	19
Figura 5. Niveles de SQuaRE.....	23
Figura 6. Estructura general de un modelo de calidad software.....	24
Figura 7. Calidad del producto norma ISO/IEC 25010	25
Figura 8. Calidad en uso norma ISO/IEC 25010.	25
Figura 9. Relación PICOC y la ecuación de búsqueda.....	37
Figura 10. Visualización de la red bibliométrica de esta investigación en VOSviewer	40
Figura 11. Instrumento para evaluar la calidad de los trabajos preseleccionados.....	43
Figura 12. Clúster nivel 1 generado por Carrot2	46
Figura 13. Clúster nivel 2 para nodo “Evaluation Model (20)” generado por Carrot2.....	47
Figura 14. Diagrama de clases para extracción de datos.....	51
Figura 15. Interfaz Web del instrumento - árbol	52
Figura 16. Interfaz Web del instrumento – opciones adicionales.....	52
Figura 17. Fases del protocolo desarrolladas	54
Figura 18. Cantidad de artículos por país de origen del autor	55
Figura 19. Ventana de observación de los estudios seleccionados	55
Figura 20. Prueba All and Some	58
Figura 21. Relación Todo - Parte	58
Figura 22. Categorías de primer nivel de TQACA	59
Figura 23. Características claves del modelo de calidad – Primer nivel de la taxonomía .	99
Figura 24. Tipos de clasificación para atributos	99
Figura 25. Modelo de calidad TQACA.....	100
Figura 26. Vista usuario	101
Figura 27. Vista proveedor.....	102
Figura 28. Vista desarrollador	103
Figura 29. Instrumento validación del modelo.....	104
Figura 30. Cálculo de la muestra con tamaño de población desconocida	106
Figura 31. Puntos de acceso para coordinar aplicación instrumento.....	107

Figura 32. Cálculo Posición cuartiles, grupo par de datos.....	110
Figura 33. Cuartiles vista desarrollador.....	111
Figura 34. Cálculo Posición cuartiles, grupo impar de datos.....	112
Figura 35. Fórmula para realizar la interpolación en el cálculo del valor del cuartil	112
Figura 36. Calculo valor Q1 con interpolación.....	113
Figura 37. Cuartiles vista usuario.....	113
Figura 38. Cuartiles vista proveedor	116
Figura 39. Guía vista desarrollador	118
Figura 40. Guía vista usuario	119
Figura 41. Guía vista proveedor.....	119
Figura 42. Clúster nivel 1 generado por Carrot2	180

LISTA DE TABLAS

Tabla 1. Áreas de prioridad cloud computing	5
Tabla 2. Descripción de fases del método	8
Tabla 3. Distribución por años de estudios	14
Tabla 4. Organismos internacionales de normalización	21
Tabla 5. Estándares relacionados con la calidad del software	21
Tabla 6. Calidad de los datos.....	26
Tabla 7. Resumen hallazgos trabajos relacionados	27
Tabla 8. Sistemas de organización del conocimiento.....	31
Tabla 9. Principios de los vocabularios controlados.....	32
Tabla 10. Criterios PICOC para la definición de las preguntas de investigación	35
Tabla 11. Repositorios digitales seleccionados.....	35
Tabla 12. Palabras claves.....	36
Tabla 13. Filtros adicionales para refinar las búsquedas.....	36
Tabla 14. Filtro por trabajos duplicados	37
Tabla 15. Ecuaciones de búsqueda personalizadas por repositorio.....	38
Tabla 16. Clústeres definidos por VOSviewer	41
Tabla 17. Trabajos preseleccionados automáticamente	41
Tabla 18. Criterios para la preselección de trabajos	42
Tabla 19. Trabajos preseleccionados manualmente	42
Tabla 20. Comparación procesos	43
Tabla 21. Trabajos que aprobaron la evaluación de calidad	44
Tabla 22. Características propuestas por el SMI.....	44
Tabla 23. Artículos del nodo “Evaluation Model (20)”	47
Tabla 24. Relación entre subnodos. Artículos nodo “Evaluation Model (20)”.....	49
Tabla 25. Atributos de “Accountability”	61
Tabla 26. Atributos de “Agility”	66
Tabla 27. Atributos de “Assurance”	72
Tabla 28. Atributos de “Empathy”	79
Tabla 29. Atributos de “Financial”	81
Tabla 30. Atributos de “Performance”	83
Tabla 31. Atributos de “Security and Privacy”	88

Tabla 32. Atributos de “Usability”	95
Tabla 33. Atributos de “User experience”	97
Tabla 34. Lista de opciones instrumento validación	105
Tabla 35. Lista de opciones instrumento validación – contexto academia.....	105
Tabla 36. Valores variables	106
Tabla 37. Aplicación del instrumento contexto industria.....	108
Tabla 38. Aplicación del instrumento contexto academia.....	108
Tabla 39. Resultados vista desarrollador	109
Tabla 40. Posición de cuartiles vista desarrollador	110
Tabla 41. Resultados vista usuario	111
Tabla 42. Posición de cuartiles vista usuario	112
Tabla 43. Valor de cuartiles vista usuario	113
Tabla 44. Resultados vista proveedor.....	114
Tabla 45. Posición de cuartiles vista desarrollador	116
Tabla 46. Descriptores de la guía para la selección de atributos de calidad	117
Tabla 47. Relación objetivos, cumplimiento y conclusiones	120
Tabla 48. Conclusiones respecto a las hipótesis.....	122
Tabla 49. Nodos de clúster nivel 1 que relacionaron solamente 2 trabajos	181

RESUMEN

El concepto de computación en la nube puede definirse como una evolución tecnológica. Entre las principales evidencias históricas de esta afirmación se encuentran: primero, el uso que empresas y universidades hacían de las grandes computadoras centrales, lo que hoy en día se conoce como servicio de accesibilidad remota a los recursos de procesamiento de otra computadora y segundo, la aparición de las redes privadas virtuales. Los servicios ofrecidos a través de esta topología de red se extrapolan a los servicios que hoy en día se brindan a través de la computación en la nube. Gracias a los servicios de la computación en la nube la tendencia actual está orientada a contratar servicios externos en centros de datos ubicados en cualquier parte del mundo.

Esta investigación aborda la temática de calidad en computación en la nube desde la perspectiva de atributos de calidad, entendidos como requisitos no funcionales que se deben atender para garantizar la satisfacción de un cliente. La línea de base de la investigación está soportada en el estándar ISO/IEC 25000:2014 y en el trabajo realizado por el consorcio para la medición de servicios en la nube, creado por la Universidad Carnegie Mellon.

Al realizar el despliegue de una aplicación en la nube, los proveedores definen características que son integradas en los acuerdos de nivel de servicio, sin embargo, estos acuerdos excluyen atributos que pueden ser claves para un desarrollador de software. La presente tesis realiza un aporte al conocimiento en la temática de atributos de calidad, definiendo una taxonomía y un modelo de calidad, que permitirá a los desarrolladores identificar qué atributos de calidad se deben tener en cuenta en una especificación de requerimientos.

En primera instancia se realizó un análisis de estándares y mejores prácticas utilizados en la construcción de taxonomías y vocabularios controlados. Posteriormente se realizó un análisis sistemático de literatura para obtener las fuentes primarias de la investigación. Se utilizaron herramientas de procesamiento automático y se realizó el contraste con un procesamiento manual. La construcción de la taxonomía está fundamentada en las sugerencias y recomendaciones de la ANSI/NISO y el Centro de Interoperabilidad Semántica de Europa. A partir de la taxonomía, se desarrolló un modelo que permitió su validación, en un contexto focalizado en desarrolladores de software con diferentes niveles de experticia.

Entre los resultados de la investigación están: 1) El protocolo para la creación de la taxonomía sobre atributos de calidad en aplicaciones diseñadas para ser desplegadas en la nube. 2) La taxonomía jerárquica y las relaciones entre los términos. 3) El modelo de

calidad como una extensión de la taxonomía y 4) Los hallazgos encontrados tras la aplicación del modelo a desarrolladores.

En esta tesis se presenta una contribución a la organización del conocimiento sobre atributos de calidad en aplicaciones diseñadas para ser desplegadas en la nube, soportada en una garantía literaria y en un proceso metodológico riguroso que puede ser verificado por terceras partes.

Palabras claves: Atributos de calidad, métricas de calidad, taxonomía, computación en la nube.

ABSTRACT

More than a novel concept, cloud computing is a technological evolution. Its origins date back to the use of mainframes by large companies and universities, which provided isolated multiuser service to remote users. Later came virtual private networks which provide isolated networking services running on top of a shared infrastructure and used to share data and resources in a safe environment within a company or organization. Finally, the search for cost reduction and greater efficiency in resource management led to moving these services to the cloud. Thanks to cloud computing the current trend is for organizations to lease external services in data centers located anywhere in the world.

The objective of this research is to explore, from the perspective of its quality attributes, the issues of software systems running in cloud environments. These attributes are linked to non-functional requirements that need to be addressed to ensure customer satisfaction. The baseline of the research is supported by the ISO/IEC 25000:2014 Standard and the work done by the Cloud Service Measurement Consortium, created by Carnegie Mellon University.

When deploying a cloud application, cloud providers define features that are integrated into service-level agreements. However, these agreements may exclude attributes that are important to software developers, final users and providers. This thesis contributes to the knowledge in the subject of quality attributes, definition of taxonomy, and quality models. This will allow developers to identify what quality attributes should be considered in the requirements specification.

This work used the following methodology: Initially, we carried out an analysis of standards and best practices used in the construction of taxonomies and controlled vocabularies. Subsequently, we performed a systematic analysis of the literature in order to obtain the primary sources of the investigation. Automated processing tools were used and compared against manual processing with the purpose of identifying pertinent works. The construction of the taxonomy is based on the suggestions and recommendations of the ANSI/NISO and the Semantic Interoperability center of Europe. The obtained taxonomy was validated using a quality model, evaluated by software developers with different levels of expertise.

Among others, the research produced the following results:

- The protocol for the creation of the taxonomy on quality attributes for cloud applications.
- The hierarchical taxonomy and the relations between the terms.
- The quality model as an extension of the taxonomy.
- The ranking of the quality attributes obtained from the application of the model by the developers.

This thesis presents a contribution to the knowledge of the classification of quality attributes in cloud applications, supported by peer reviewed literature and a rigorous methodological process that can be verified by third parties.

Keywords: Quality attributes, quality metrics, taxonomy, cloud computing.

1. INTRODUCCIÓN

1.1. Planteamiento del problema

El estudio previo Guerrero y Londoño (2016) y el análisis del concepto computación en la nube permite obtener tres conclusiones; en primera instancia se establece que la computación en la nube es un conjunto de servicios que son ofrecidos desde hace muchos años a través de las redes de comunicaciones; en segunda instancia, es la evolución de los servicios que se prestan hace más de una década desde Internet y finalmente, la computación en la nube es una estrategia comercial de las grandes empresas, que han invertido elevadas sumas de dinero para ofrecer soluciones de valor y obtener dividendos, gracias a la infraestructura que poseen hoy en día.

La evolución del concepto de computación en la nube, define un objetivo común: permitir a las empresas el uso de sus aplicaciones de manera eficiente, con una capacidad de administración mejorada que permita a los departamentos de tecnologías de la información (TI), ajustar rápidamente los recursos requeridos de acuerdo a una demanda empresarial fluctuante (Oestreich, 2013). Los recursos son provistos por los proveedores, quienes son los encargados de permitir el uso de las aplicaciones y son ellos los que desarrollan y despliegan servicios que pueden ser reutilizados por muchas aplicaciones. De esta forma, el cliente final puede reutilizar servicios e incluso construir nuevas aplicaciones con base en lo que ofertan los proveedores (Oh, La, & Kim, 2011). En consecuencia, el cumplimiento de los requerimientos no funcionales pasa a ser una responsabilidad compartida entre diferentes actores como proveedores de servicios en la nube (IaaS), proveedores de red, proveedores de servicios (PaaS), entre otros, a diferencia de esquemas tradicionales donde todos los aspectos están bajo el control de uno o pocos actores.

El gran impacto que la computación en la nube ha tenido sobre la industria, es un área de estudio e investigación. Para Singh y Chana (2012) los temas más relevantes en computación en la nube son: a) arquitectura basada en la nube, b) seguridad, c) desarrollo basado en componentes y reusabilidad, d) calidad en el servicio y e) el diseño. Un estudio realizado por Guerrero y Londoño (2016), confirmó la importancia de los temas mencionados y amplió las temáticas de la siguiente forma: a) infraestructura como servicio, b) calidad, c) seguridad, d) conceptualización, e) software como servicio, f) casos aplicados, g) acuerdos de niveles de servicio y h) desarrollo de aplicaciones móviles. Según Guerrero y Londoño (2016) el segundo tema en importancia en computación en la nube es la calidad.

El concepto de calidad está definido en el estándar Std 610.12-1990 como “el grado en el que un sistema, componente o proceso, cumple con los requerimientos especificados y las necesidades o expectativas del cliente o usuario” (IEEE, 1990). La norma ISO 9001:2015 la define como “el grado en el que un conjunto de características inherentes

cumple con los requisitos” (ISO, 2015). Estas definiciones no son ajenas a la computación en la nube, por el contrario, se evidencia en la subclasificación sobre calidad propuesta por Guerrero y Londoño (2016): a) Calidad del servicio, b) calidad en el proceso de desarrollo, c) calidad computación móvil en la nube, d) calidad de los datos y e) calidad de la experiencia.

En el estudio realizado por Guerrero y Londoño (2016), se analizan las referencias sobre calidad en computación en la nube: (Alnanih, Ormandjieva, & Radhakrishnan, 2013), (Franke, Kowalewski, & Weise, 2012), (Hammam & Senbel, 2013), (Hoang, Niyato, & Wang, 2012), (La & Kim, 2013), (Hobfeld, Schatz, Varela, & Timmerer, 2012), (Comerio, Truong, Batini, & Dustdar, 2010), (Klems, Bermbach, & Weinert, 2012), (Yang & Jia, 2011), (Zrnec & Lavbič, 2011), (Bao, Xiao, Sun, & Zhao, 2010), (Belalem, Bouamama, & Sekhri, 2011), (Boniface et al., 2010), (Du & Li, 2013), (Garg, Versteeg, & Buyya, 2013), (Hershey, Rao, Silio, & Narayan, 2012), (Hsueh, Wang, & Ting, 2012)(Hu & Zhang, 2013), (C. Huang, Hsu, & Tzeng, 2012), (J. Huang & Yanbing, 2013), (Hussain & Abdulsalam, 2014), (Kiruthika, Horgan, & Khaddaj, 2012), (Krebs, Momm, & Kounev, 2013), (N. Kumar, Chilamkurti, Zeadally, & Jeong, 2013), (P. Kumar, Sehgal, & Diwakar, 2011), (X. Li & Yu, 2011), (Z. Liu, Liu, Lu, Cai, & Yang, 2010), (Marinescu & Petcu, 2013), (Massonet & Arenas, 2012), (Qian, Medhi, & Trivedi, 2011), (Reynolds, Hopkinson, Oxley, & Mullins, 2011), (Sagbo & Houngue, 2012), (Song, Zhang, Gong, & Dai, 2012), (Sun, Chang, Miao, & Wang, 2013), (Shangguang Wang, Liu, Sun, Zou, & Yang, 2012), (Shou-xin Wang, Zhang, Wang, & Qiu, 2010), (You et al., 2011), (Z. Zheng, Wu, Zhang, Lyu, & Wang, 2013), (Chatzipetrou, Angelis, Barney, & Wohlin, 2011), (Chauhan & Babar, 2012), (R Colomo-Palacios & Soto-Acosta, 2011), (Grbac & Huljenic, 2011), (Guha & Al-Dabass, 2010), (Hashmi et al., 2011), (Hou & Zhou, 2010), (Miyamoto, Nerome, & Nakamura, 2012), (Patidar, Rane, & Jain, 2011), (P. Raj, Venkatesh, & Amirtharajan, 2013) (P. Raj et al., 2013), (Richardson, Casey, McCaffery, Burton, & Beecham, 2012) y (Salger, Engels, & Hofmann, 2010), este estudio permitió inferir los siguientes hallazgos e interrogantes:

- A pesar de la existencia de atributos de calidad y de modelos para medir la calidad del servicio, difícilmente se pueden aplicar a proveedores privados, esto implica que exista para el cliente un modelo de caja negra al momento de establecer los acuerdos de niveles de servicio. ¿Son necesarios todos los atributos de calidad para medir la calidad del servicio?, ¿Por qué algunos estudios analizan algunos atributos y no todos?
- Los estudios utilizan una terminología similar, no obstante, difieren en conceptos que pueden inducir al error. Los estudios utilizan diferentes enfoques, sin embargo, entre un enfoque y otro un concepto varía desde su concepción, hasta en la forma como se obtienen resultados. ¿Un cuerpo de conocimiento más completo y unificado podría servir de base para futuras investigaciones?

- Los modelos analizados en el estudio de Guerrero y Londoño (2016) están orientados a proveedores públicos en procesos de prestación de servicio ideal. La computación en la nube está dominada por 6 proveedores mundiales, los cuales están en mora de proponer un único modelo que garantice a los usuarios la calidad del servicio que se exige. ¿Un modelo estructurado podría servir de base para analizar qué debe ofrecer un proveedor?
- Las metodologías de desarrollo tradicionales están siendo adaptadas a procesos de desarrollo para la computación en la nube, sin embargo, aún no es clara la participación de los proveedores en los procesos de desarrollo orientados a la nube.
- La computación móvil es uno de los principales clientes de los servicios provistos en la nube, por lo tanto, la medición de requerimientos no funcionales impacta de manera directa los requerimientos funcionales, enfocando los desarrollos dentro de los contextos de ecosistemas digitales. ¿Qué requerimientos funcionales se deben tener en cuenta? ¿Cómo se mide un requerimiento no funcional si la gran mayoría de estudios lo hace de una manera diferente?

La Figura 1 resume lo planteado en esta sección, e identifica los problemas actuales y causas de la ausencia de estandarización de criterios de calidad en computación en la nube.

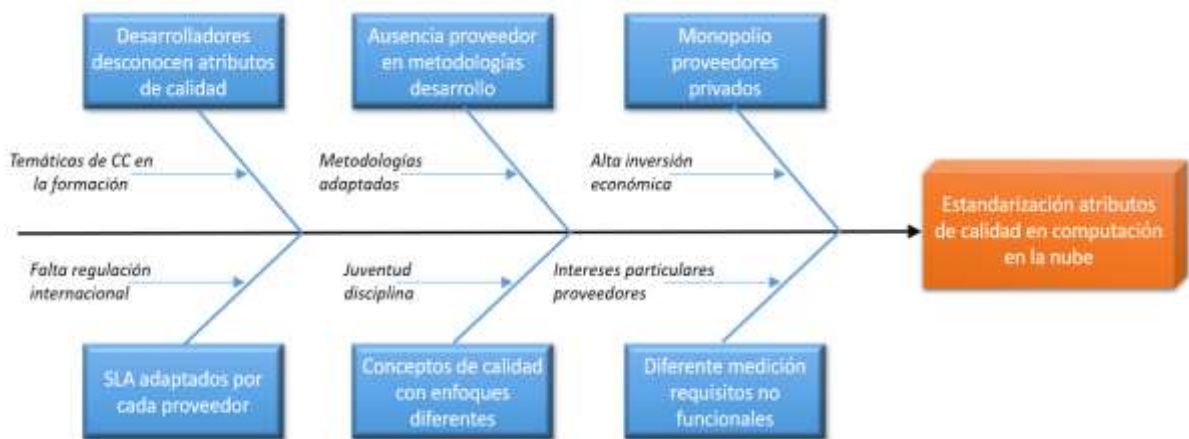


Figura 1. Espina-pescado planteamiento problema

La organización del conocimiento en cualquier disciplina es una tarea compleja que sirve para establecer líneas de base en futuras investigaciones. La juventud de la computación en la nube brinda una oportunidad para realizar contribuciones en los cimientos de esta disciplina, siempre y cuando exista una articulación entre las bases conceptuales y los estudios generados al alrededor de esta disciplina.

Entre los hallazgos enunciados surge una oportunidad para realizar un aporte en la organización del conocimiento en la temática de calidad en computación en la nube,

mediante la construcción de una taxonomía, que permita definir la estructura jerárquica de los términos en el tema atributos de calidad en aplicaciones diseñadas para ser desplegadas en la nube. El producto final de este proyecto entregará un valor agregado en el contexto de calidad en computación en la nube, puesto que busca la formalización y estandarización de términos, que podrían emplearse en contextos académicos, de investigación y en la industria del software.

1.2. Objetivos de la investigación

1.2.1. Objetivo general

Construir una taxonomía soportada en un vocabulario controlado, que permita definir una estructura jerárquica de términos y sus relaciones en diferentes dimensiones, para realizar una contribución a la organización del conocimiento sobre atributos de calidad en aplicaciones diseñadas para ser desplegadas en la nube.

1.2.2. Objetivos específicos

1. Comparar los estándares y mejores prácticas para la construcción de vocabularios controlados y taxonomías, que permita determinar el esquema de etiquetado, clasificación e indexación.
2. Diseñar un protocolo para crear un vocabulario controlado sobre atributos de calidad en aplicaciones diseñadas para ser desplegadas en la nube, de acuerdo con las mejores prácticas y estándares definidos en la literatura científica para la construcción de vocabularios.
3. Evaluar estudios sobre la creación y uso de vocabularios controlados, taxonomías, tesauros y ontologías en computación en la nube, para identificar la articulación del aporte del presente estudio en la organización del conocimiento.
4. Construir un vocabulario controlado sobre atributos de calidad, que permita organizar el conocimiento recopilado en la temática y soporte el desarrollo de la taxonomía, utilizando el protocolo diseñado en el segundo objetivo específico de la presente propuesta.
5. Construir a partir de la taxonomía creada, un modelo de calidad compuesto por métricas, que pueda ser utilizado en la fase de especificación de requerimientos de un proceso de desarrollo software.
6. Validar la taxonomía propuesta mediante la utilización del modelo de calidad construido en el objetivo anterior, en dos contextos: industria software y academia.

Para analizar el impacto de la taxonomía en proyectos de desarrollo, bajo la supervisión de un panel de expertos.

1.3. Justificación

Un vocabulario controlado permite establecer una abstracción de conceptos y sus relaciones (NIST, 2011a). Sirve para establecer lineamientos que permitan la correcta utilización de los conceptos y son la línea de base para construir taxonomías, tesauros y ontologías. El NIST estableció una taxonomía de cuatro niveles para la computación en la nube. En el primer nivel define las responsabilidades de los interesados, el segundo nivel define las actividades de cada uno de los roles, el tercer nivel es un componente que define las tareas que se requieren para ejecutar una acción, y finalmente, el cuarto nivel es un subcomponente del nivel anterior que permite especificar detalles acerca de la portabilidad.

Tal como lo presenta NIST, el principal objetivo de la computación en la nube es brindar un método económico que proporcione servicios de alta calidad, de forma rápida a un menor costo (NIST, 2011b). En el mapa de ruta de los estándares de computación en la nube propuesto por el NIST, el Departamento de Defensa (DoD), definió las áreas con brechas de estandarización que tienen alta prioridad, ver Tabla 1.

Tabla 1. Áreas de prioridad cloud computing

Áreas	Detalles del área
Interfaces funcionales SaaS ¹	<ul style="list-style-type: none"> ✓ Formato de datos y estándares de interfaz para la productividad del correo electrónico ✓ Formatos de metadatos y estándares de interfaz
Interfaces para la administración de autoservicio SaaS	<ul style="list-style-type: none"> ✓ Estándares de interfaz relacionados con cuentas de usuario y credenciales
Interfaces funcionales PaaS ²	<ul style="list-style-type: none"> ✓ Estándares para el formato de los datos buscando la serialización y deserialización
Soporte comercial, aprovisionamiento y configuración	<ul style="list-style-type: none"> ✓ Estándares para acuerdos de nivel de servicio y la calidad de los servicios ✓ Estándares para describir y descubrir servicios ✓ Estándares para la medición y facturación del consumo de los servicios
Seguridad	<ul style="list-style-type: none"> ✓ Estándares para el aprovisionamiento y administración de identidades en diferentes dominios de red ✓ Estándares y políticas de acceso a través de diferentes sistemas ✓ Estándares de inicio de sesión único con protocolos de

¹ SaaS: Software as a Service

² PaaS: Platform as a Service

	autenticación fuerte
Accesibilidad	✓ Marcos de trabajo estandarizados para el intercambio de requerimientos de accesibilidad

Fuente: NIST Cloud Computing Reference Architecture. (NIST, 2011b)

Como se puede apreciar en la Tabla 1, una de las brechas prioritarias identificadas por el DoD, está relacionada con la “definición de atributos de calidad del servicio”. Sobre este tema, el United States Government (USG) realizó la siguiente recomendación: “Acelerar el desarrollo y uso de estándares en computación en la nube” (NIST, 2011b).

Existen otros esfuerzos relacionados con la organización del conocimiento en computación en la nube, Polash et al. (2014) proponen una comparación de taxonomías para computación en la nube en donde se presentan tres tópicos claves: conceptual, rendimiento y seguridad. En las 15 taxonomías analizadas por Polash et al. (2014), algunos autores definen conceptos claves de computación en la nube, otros brindan herramientas para comparar servicios, proveedores, entre otros. Los esfuerzos realizados en esta organización del conocimiento no solo son realizados por investigadores, también existen asociaciones como OpenCrowd (OpenCrowd, 2010), que buscan ampliar el dominio del conocimiento y proponer soluciones en áreas como rendimiento, computación verde y seguridad. No obstante, la mayoría de los esfuerzos están enfocados en SaaS (Hou y Zhou, 2010), IaaS³ (Guo, 2003) y PaaS (Ardagna et al., 2012), otros están enfocados en el proveedor, el cliente y el intermediario entre proveedor y el cliente, muy pocos tienen un objetivo específico que permita definir bases conceptuales para nuevos trabajos de investigación.

El tema de atributos de calidad en computación en la nube es un tema de prioridad para el USG, sin embargo, para la comunidad científica es un tema que aún no ha sido explorado en su totalidad. ¿Por qué estos temas aún no han sido explorados? La respuesta puede derivar de dos factores: por un lado, la evolución permanente de la computación en la nube durante los últimos 8 años y por el otro, la tipificación de tópicos de vital importancia para los proveedores, como es el caso de la seguridad.

1.4. Método de investigación

Para el desarrollo de la tesis doctoral se utilizó la aplicación del enfoque de investigación mixto (Hernández-Sampieri, Fernández-Collado, & Baptista-Lucio, 2014). Por un lado, el método cuantitativo representa un conjunto de procesos de tipo secuencial y probatorio, se basa en investigaciones previas y se utiliza para consolidar los conceptos y establecer con exactitud patrones de comportamiento; en segunda instancia el método cualitativo permite que el investigador se forme conceptos propios sobre el fenómeno estudiado

³ IaaS: Infrastructure as a Service

partiendo de una realidad por descubrir, construir e interpretar. Las actividades del método se desarrollaron en fases representadas en la Figura 2.



Figura 2. Fases del método

En la Tabla 2 se describen las actividades desarrolladas en cada fase.

Tabla 2. Descripción de fases del método

Fase 1: Selección de estándares y mejores prácticas	
Descripción	En esta fase se analizaron los conceptos presentados por Heather Hedden en “The Accidental Taxonomist” (Hedden, 2010), posteriormente, la conferencia realizada por Alice Redmond-Neal en SLA Annual Conference 2013 titulada “Starting a Taxonomy Project: Taxonomy Basics” (Redmond-Neal, 2013). El análisis continuó con la revisión de la guía Z39.19-2005 publicada por ANSI/NISO (ANSI/NISO, 2005) y finalmente se analizaron los lineamientos propuestos por el Centro de Interoperabilidad Semántica Europa en (Fraunhofer ISST, 2009)
Actividades / Herramientas	<p>Lectura comparativa de los documentos de la línea de base:</p> <ul style="list-style-type: none"> ✓ The Accidental Taxonomist” (Hedden, 2010) ✓ Starting a Taxonomy Project: Taxonomy Basics” (Redmond-Neal, 2013) ✓ guía Z39.19-2005 publicada por ANSI/NISO (ANSI/NISO, 2005) ✓ Lineamientos del Centro de Interoperabilidad Semántica Europa en (Fraunhofer ISST, 2009) <p>Herramientas utilizadas: Word, Excel y XMind.</p>
Producto	<ol style="list-style-type: none"> 1. Lineamientos para la construcción de la taxonomía y el protocolo 2. Protocolo para la creación de un vocabulario controlado sobre atributos de calidad en aplicaciones diseñadas para ser desplegadas en la nube.
Fase 2: Ejecución del protocolo	
Descripción	En esta fase se realizó la búsqueda y procesamiento de estudios relacionados con atributos de calidad en computación en la nube. Los repositorios consultados fueron: IEEE Xplore, Scopus, Science Direct y ACM.
Actividades / Herramientas	<ul style="list-style-type: none"> ✓ Calibración y ejecución de la cadena de búsqueda ✓ Búsquedas en Repositorios IEEE Xplore, Scopus, ACM y Science Direct ✓ Depuración de trabajos, eliminación de repetidos ✓ Procesamiento automático por medio de VOSviewer ✓ Procesamiento manual <p>La revisión manual de los trabajos se realizó por 3 metadatos: título, resumen y palabras claves.</p> <p>Herramientas utilizadas: Repositorios digitales, VOSviewer, Word y Excel.</p>

Fase 2: Ejecución del protocolo	
Producto	<ol style="list-style-type: none"> 1. Cadenas de búsqueda para la selección de estudios. 2. Listado de trabajos preseleccionados.
Fase 3: Procesamiento de información automático y manual	
Descripción	<p>En esta fase se clasificaron los estudios preseleccionados en clústeres, se realizó la evaluación de calidad de los trabajos preseleccionados con el objetivo de obtener el listado de documentos primarios para este estudio.</p>
Actividades / Herramientas	<p>Evaluación de calidad de los trabajos preseleccionados luego de los procesamientos. Uso de Carrots2 para encontrar los clústeres y Extracción de datos.</p> <p>Las entradas de la etapa de extracción de datos fueron:</p> <ul style="list-style-type: none"> ✓ Las características propuestas por el SMI ✓ Los clústeres generados por Carrot2 y ✓ Los trabajos que aprobaron la fase de evaluación de calidad. <p>Herramientas utilizadas: Carrot2, aplicación web para extracción de datos, instrumento en Excel construido con base en la metodología PICOC.</p>
Producto	Estudios seleccionados que conforman las fuentes primarias de la tesis.
Fase 4: Construcción de la taxonomía	
Descripción	<p>Se realizó la taxonomía TQACA siguiendo las recomendaciones de (ANSI/NISO, 2005) y de (Fraunhofer ISST, 2009). El resultado fue una clasificación de características de calidad organizadas en forma jerárquica.</p>
Actividades / Herramientas	<ul style="list-style-type: none"> ✓ Clasificación de atributos de calidad ✓ Establecimiento de relaciones ✓ Reutilización de la propuesta de SMI para la estructura jerárquica de la taxonomía. <p>Herramientas utilizadas: Aplicación web para presentación de la estructura jerárquica, Excel.</p>
Producto	Taxonomía de atributos de calidad para computación en la nube.

Fase 5: Construcción del Modelo de calidad

Descripción

En esta fase se construye el modelo de calidad que busca proveer un instrumento que pueda ser utilizado por desarrolladores en la fase de requerimientos. Al ser una extensión de la taxonomía TQACA realizada en la presente investigación, reutiliza la totalidad de las características y atributos clasificados en 3 vistas: desarrollador, proveedor y usuario.

Actividades / Herramientas

Para la construcción del modelo se realizaron las siguientes actividades:

- ✓ Definición de las vistas por actores
- ✓ Desarrollo de instrumentos para la definición del ranking de atributos de calidad

Herramientas utilizadas: Excel, plataforma web.

Producto

Modelo de calidad con 3 vistas: usuario, proveedor y desarrollador.

Fase 6: Evaluación del Modelo de calidad

Descripción

En el proceso de validación se utilizó el instrumento desarrollado para la extracción de datos, adicionando una extensión que le permitiera al desarrollador identificar los atributos de calidad susceptibles a ser utilizados en una fase de requerimientos, clasificados desde 3 vistas: usuario, proveedor y desarrollador.

Actividades / Herramientas

Las actividades de esta fase fueron:

- ✓ Definición de la muestra por medio de tamaño de población infinita
- ✓ Definición del ranking con base en Calculo posición cuartiles
- ✓ Aplicación del instrumento en Excel para la evaluación del uso de los atributos en el grupo de desarrolladores seleccionados.

Herramientas utilizadas: Word, Excel, plataforma web.

Producto

Guía para la selección de atributos de calidad en la etapa de requerimientos.

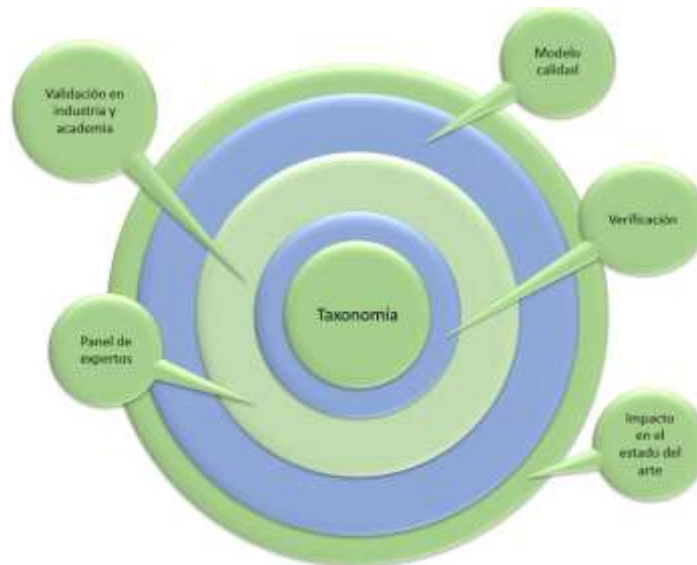
1.5. Alcance de la tesis

La presente propuesta está sustentada en una metodología que apoya la generación de conocimiento con el desarrollo de cada una de sus fases. Inicialmente se realizará la construcción de un protocolo para la creación de una taxonomía sobre atributos de calidad en aplicaciones diseñadas para ser desplegadas en la nube. Posteriormente se realizará la definición de grupos de conceptos sobre atributos de calidad. Cada agrupación tendrá la articulación de los conceptos encontrados de conformidad con los estudios analizados y se realizará la definición del mapa de relaciones entre conceptos, así como el establecimiento de la estructura jerárquica de la información. El producto final de nuevo conocimiento será la creación de una taxonomía sobre atributos de calidad en aplicaciones diseñadas para ser desplegadas en la nube.

El tipo de innovación del presente proyecto se puede catalogar como innovación de producto y de proceso (OCDE - Eurostat, 2007), que a nivel de producto se obtendrá la creación de una taxonomía sobre atributos de calidad en aplicaciones diseñadas para ser desplegadas en la nube, las mejoras significativas en la organización del conocimiento en este ámbito, radican en las clasificaciones que serán generadas y un elemento novedoso no incluido en otros estudios como: la perspectiva de los actores principales de un proceso de desarrollo para la nube: proveedor, desarrollador y cliente. Este trabajo busca minimizar una de las brechas identificadas por el DoD con relación a la “buena definición de atributos de calidad del servicio”. A nivel de innovación de proceso, se definirá un método o proceso mejorado, que permitirá la construcción de la taxonomía enunciada.

En la Figura 3 se puede apreciar el alcance de la propuesta en forma de anillos. La base del proyecto es la construcción de la taxonomía, a través de la definición de un vocabulario controlado que permita organizar el conocimiento en la temática de atributos de calidad. Con respecto a la verificación de la taxonomía, se plantea el uso de la técnica denominada “cross-validation”, útil para analizar el resultado de un proceso estadístico, de tal manera que se garantice una división entre los datos, en este caso, los datos que se usarán para la construcción de la taxonomía y los datos para la prueba de verificación. La fase de validación incluye elementos como la utilización de la taxonomía en procesos de desarrollo en el contexto de la industria software y la academia, realizando una clasificación tipos de proyectos de baja, media y gran envergadura, además del acompañamiento de un panel de expertos que emitirá un juicio de pertinencia sobre el trabajo realizado. Otro aporte del proyecto es la generación de un modelo de calidad, que podrá ser utilizado en procesos de desarrollo orientados a la nube, particularmente en las fases de requerimientos y verificación, todo esto buscando analizar el verdadero impacto del proyecto en el estado del arte del área abordada.

Figura 3. Alcance de la tesis



1.6. Contribuciones de la tesis

La investigación presenta 4 contribuciones clasificadas como generación de nuevo conocimiento:

1. Protocolo para la creación de taxonomías: A partir del proceso propuesto por (Kitchenham, 2007), se construyó un protocolo que presenta los pasos detallados para realizar la revisión de literatura y el procesamiento de la información bajo dos estrategias: automática y manual. Los resultados de la investigación muestran que el margen de error entre el procesamiento automático de la información y el procesamiento manual es bajo.
2. Organización del conocimiento: La creación de una taxonomía sobre atributos de calidad para aplicaciones a ser desplegadas en la nube, presenta una nueva perspectiva sobre la organización de las características de calidad que pueden ser medidas en la computación en la nube. La taxonomía es un instrumento que puede facilitar la transferencia de conocimiento en procesos de formación universitaria, además, sirve como base para nuevas investigaciones en la temática de estudio.
3. Instrumento para apoyar la creación de taxonomías: Una herramienta orientada a la web para apoyar la construcción de taxonomías jerárquicas, permite realizar el registro de información, para construir una taxonomía que pueda visualizarse en la web. Se constituye en un instrumento útil en procesos de investigación, con características adicionales a herramientas como Carrot2 y Atlas.ti.
4. Modelo de calidad: El modelo de calidad es una extensión de la taxonomía que define los atributos de calidad desde varias perspectivas, permitiendo a un desarrollador, tener un conjunto de atributos de calidad susceptibles a ser utilizados en una especificación de requerimientos para una aplicación que será desplegada en la nube.

1.7. Estructura del documento

En el capítulo 1 se describe la problemática que fundamenta la motivación del autor para realizar el trabajo de investigación. Se referencian los objetivos de la tesis, derivados de un análisis de estudios previos y la justificación. Se explica el método de investigación, el alcance de la tesis y finalmente se enumeran las contribuciones del trabajo realizado.

En el capítulo 2 se presentan los trabajos relacionados con la presente investigación. Inicialmente aborda la calidad en la computación en la nube desde la perspectiva de taxonomías realizadas. Posteriormente presenta un análisis de trabajos relacionados con atributos de calidad desde la perspectiva de modelos de calidad. Continúa con la presentación del índice de medición del servicio y finaliza con la literatura relacionada con estándares y normatividad.

El capítulo 3 contiene una revisión literaria sobre vocabularios controlados, un enfoque que permite diferenciar los diferentes sistemas de organización del conocimiento y define los principios básicos de los vocabularios controlados. Introduce el concepto de garantía literaria, el cual es fundamental en la presente investigación.

En el capítulo 4 se describe el protocolo diseñado y ejecutado para la construcción de la taxonomía. Plantea la necesidad, la planeación y la ejecución del protocolo. Adicionalmente, se presenta un resumen que le permite al lector tener una visión clara de todo el proceso realizado. Al finalizar el capítulo se encuentran las amenazas a la validez de la ejecución del protocolo.

En el capítulo 5 se presenta la taxonomía. Se hace una introducción al cumplimiento de los principios básicos para la construcción de taxonomías y se ubica al lector en el tipo de taxonomía que se desarrolló. También se describen las características y los trabajos relacionados para cada uno de los hallazgos.

El capítulo 6 describe el modelo de calidad como una extensión de la taxonomía, el instrumento aplicado para conocer los atributos de calidad más utilizados en cada vista. Adicionalmente, se presenta la distribución de los puntos de acceso donde se realizó la validación del modelo y los resultados obtenidos.

2. TRABAJOS RELACIONADOS

La revisión acerca de la temática de computación en la nube se organizó en 4 perspectivas: taxonomías existentes, modelos de calidad implementados, índice de medición del servicio en la nube y normatividad legal alrededor del tema de computación en la nube. Cada perspectiva fue incluida con el objetivo de establecer claridad en los temas relacionados en calidad en computación en la nube.

2.1. Calidad en computación en la nube

Identificar qué taxonomías se han modelado o construido aporta una base de conocimiento acerca de características, dimensiones y usos de estas estructuras, con el fin de definir una línea de base en la construcción de la taxonomía del presente estudio.

2.1.1. Taxonomías

Se realizó una búsqueda sobre taxonomías de computación en la nube, los artículos encontrados oscilaron en un rango temporal entre enero de 2009 y junio de 2017, evidenciando la participación de la comunidad científica en el desarrollo de vocabularios controlados y taxonomías en el área de computación en la nube. Cronológicamente los trabajos se distribuyen como se aprecia en la Tabla 3.

Tabla 3. Distribución por años de estudios

Año	Cantidad	Porcentaje
2009	3	3%
2010	2	2%
2011	10	11%
2012	3	3%
2013	9	10%
2014	19	21%
2015	22	24%
2016	12	13%
2017	11	12%
Total	91	100%

De los 91 estudios que abordan la temática de taxonomías en los diferentes contextos de la computación en la nube, 7 trabajos se identificaron como taxonomías relacionadas con aspectos de calidad en la nube. A continuación, se presenta el análisis de cada uno de estos estudios.

Rimal et al., (2009) exponen un estudio sobre las diferentes características que ofrecen los proveedores de servicios de computación en la nube, tanto privados como de acceso

abierto, ofreciendo como resultado el alcance de cada uno, el cual puede ser empleado en la toma de decisiones para la adquisición de estos servicios.

Los autores Hoefler & Karagiannis (2010, 2011), describen los actuales servicios de computación en la nube a través de una taxonomía para su clasificación diseñada en forma de árbol, esto permite comparaciones rápidas, dando al usuario un conjunto de opciones en cada nivel. Esta estructura clara hace que la comparación de los servicios de computación en la nube sea más eficiente que la utilización de comparaciones basadas en tablas. Además, la taxonomía no sólo sirve para mapear servicios de computación en la nube, sino que también ayuda a los clientes potenciales y desarrolladores a señalar qué características debería tener el servicio deseado. Los principales servicios que hacen parte del estudio son: IaaS, PaaS, y SaaS, desplegando las características de calidad y servicios específicas para cada uno.

Por su parte Gonzalez et al., (2011) presentan un estudio sobre conceptos existentes y taxonomías y armoniza estos dos enfoques en un modelo de taxonomía extensible para los servicios de computación en la nube, basada en la taxonomía SPI (Software, Platform and Infrastructure) creada por el NIST, para crear una organización jerárquica que agrupa diferentes servicios según sus características. Gonzalez et al., (2011) comparan cuatro propuestas de taxonomías y concluyen que son genéricas y no proporcionan la granularidad necesaria para comparaciones más profundas y complejas entre servicios distintos. El resultado es una estructura arborescente que agrega soluciones de computación en la nube, favoreciendo la identificación de similitudes, dependencias y puntos complementarios entre los servicios SPI.

(Z. Li, O'Brien, Cai, & Zhang, 2012) introducen una taxonomía para ayudar a perfilar y estandarizar los detalles de la evaluación del desempeño de los servicios comerciales sobre computación en la nube, se concentran en características como comunicación, procesamiento, memoria, almacenamiento, velocidad de transacción, disponibilidad, latencia, confiabilidad, ancho de banda, escalabilidad y variabilidad. Su trabajo se justifica en tres principales problemas: terminología no estandarizada, análisis correcto pero impreciso y análisis incorrecto, falencias identificadas en otros estudios. La taxonomía se construye a lo largo de dos dimensiones: característica de rendimiento y experimento.

Polash et al. (2014), proponen una comparación de taxonomías para computación en la nube para apoyar a investigadores y profesionales de la academia y la industria con la organización de conceptos y terminología sobre computación en la nube.

Mustafa et al., (2015) presentan una revisión exhaustiva de las técnicas de gestión de recursos (Resource Management -RM-) y elaboran una taxonomía basada en características como conciencia de consumo energético, acuerdos de niveles de servicio, orientación al mercado, balanceo, carga de red, nube híbrida y computación en la nube

para móviles. Sobre calidad del servicio agrupa estudios referentes a acuerdos de niveles de servicio y características relevantes en este contexto.

Es importante mencionar el reporte realizado por el NIST (NIST, 2011b) como uno de los referentes principales que aunque es amplio sobre aspectos de computación en la nube, establece de manera estándar la arquitectura de referencia en esta área. De hecho, hace parte de los trabajos realizados por otros autores.

Adicionalmente, a la revisión de estudios en fuentes de alto impacto y conferencias, se realizó una revisión de tesis doctorales, encontrando que Jiménez-Domingo (2013), reunió en su tesis doctoral los conceptos más relevantes y emergentes en la actualidad como Cloud Computing, SaaS, tecnologías semánticas, modelado de procesos de negocio, interoperabilidad entre sistemas, entre otros, como base para definir un estándar o lenguaje común que permita la interoperabilidad entre aplicaciones dentro de un entorno de Cloud Computing. La solución propuesta se enmarcó en el diseño de un lenguaje semántico basado en ontologías y de un sistema global que trabaja a su alrededor capaz de capturar el conocimiento de las aplicaciones alojadas en la nube, habilitando la realización de procesos de intercambio de información entre aplicaciones heterogéneas de forma automática. Por su parte, Takabi (2013) propuso un marco de gestión de políticas basado en la semántica para entornos de computación en la nube que permite a los usuarios especificar políticas de acceso que se aplican a sus recursos utilizando un marco unificado sin importar dónde se almacenen los recursos.

Esta revisión permitió concluir que los estudios analizados evidencian la preocupación de la comunidad científica por organizar el conocimiento referente a los servicios de computación en la nube, por lo general son taxonomías que clasifican servicios o características específicas. Sin embargo, no profundizan en la organización de una taxonomía sobre atributos de calidad en computación en la nube. Por todo lo anterior, se puede expresar que el desarrollo de esta investigación será un aporte al conocimiento, teniendo en cuenta que entregará entre otros productos, el diseño de un protocolo que permitirá construir una clasificación del conocimiento sobre atributos de calidad en aplicaciones diseñadas para ser desplegadas en la nube, complementando los estudios que a la fecha se han realizado sobre taxonomías en computación en la nube.

2.1.2. Modelos de calidad

El alcance del presente estudio no solo incluye las taxonomías, también incluye la implementación de modelos de calidad, por lo tanto, es importante identificar qué trabajos han propuesto modelos de calidad en computación en la nube, con el objetivo de acumular una base de conocimiento acerca de este tópico.

En el apartado anterior se abordaron estudios sobre taxonomías de computación en la nube y se identificaron aquellas relacionadas específicamente con aspectos de calidad en

la nube. En este apartado se presentan trabajos relacionados con modelos de calidad para computación en la nube.

Lee, Lee, Cheun, & Kim (2009), proponen un modelo para evaluar la calidad de SaaS. El modelo define cinco atributos de calidad: reusabilidad, disponibilidad, escalabilidad, confiabilidad y eficiencia. El modelo se basa en las características principales de SaaS y define métricas para cada atributo. La evaluación del modelo se realiza por medio de tres criterios propuestos por el estándar IEEE Std 1061: correlación, consistencia y poder discriminativo. Este último hace referencia a una métrica capaz de discriminar componentes software de alta y baja calidad.

Salama, Shawish, Zeid, & Kouta (2012), proponen un modelo matemático con el fin de optimizar el proceso de selección de los proveedores de servicios en la nube, con base en las garantías de la calidad del servicio. Los atributos seleccionados para este modelo son: tiempo de respuesta, rendimiento y tiempo disponible. La validación del modelo se realiza a través de simulaciones.

Wen & Dong (2013), plantean un modelo de calidad que mide la seguridad, calidad del servicio y calidad de software del servicio SaaS. Esta evaluación es realizada desde tres perspectivas: de plataforma, del proveedor y del cliente. Los autores clasifican el servicio SaaS en cuatro niveles: básico, estándar, optimizado e integrado.

X. Zheng, Martin, Brohman, & Xu (2014), plantean un modelo de calidad para servicios en la nube denominado CLOUDQUAL. Este modelo contiene seis dimensiones de calidad: usabilidad, disponibilidad, confiabilidad, capacidad de respuesta, seguridad y elasticidad. Se trabaja con un caso de estudio con tres nubes de almacenamiento, los resultados demostraron que CLOUDQUAL puede evaluar la calidad de las nubes, demostrando así la efectividad del modelo. Plantea un método formal para validar el modelo, por medio de criterios estándar como correlación y consistencia, entre otros factores.

Naseer, Jabbar, & Zafar (2014) proponen un modelo orientado hacia la selección de servicios de proveedores eficientes y confiables, con base en el análisis de los datos recolectados por las autoridades regulatorias. Esta información presenta el desempeño de los proveedores y los comentarios de los clientes en un tiempo de un año. Los atributos analizados en este modelo son: medidas de seguridad, cumplimiento de las normas del organismo regulador, tiempo de inactividad, tiempo de actividad, soporte al cliente, desempeño de un servicio específico y latencia.

Zhou, Wang, Li, & Jiang (2015) proponen un modelo de calidad que abarca características TI y servicios en la nube. El modelo de calidad se construyó basado en seis características: usabilidad, seguridad, confiabilidad, tangibilidad, capacidad de respuesta y empatía. La forma de organización jerárquica del modelo divide las características en

subcaracterísticas las cuales tienen asociadas métricas objetivas y subjetivas. La evaluación del modelo se realiza de manera ponderada, es decir, se asignan pesos a las subcaracterísticas de acuerdo con el grado de importancia para el cliente. Se valida el modelo por medio de un caso de estudio de un servicio denominado Centrin Data Systems (CDS).

Z. Wang, Jiang, & Zhou (2015) plantean dos modelos para la gestión de la calidad: 1) el primer modelo está orientado al mantenimiento del servicio desde dos perspectivas cliente y proveedor. Para el cliente la calidad se define como la brecha entre la calidad esperada y la percepción del usuario sobre la calidad recibida (calidad de uso). Los proveedores por su parte son responsables por cómo entregar los servicios y los recursos a los usuarios. 2) el segundo modelo para la evaluación del modelo de calidad, el cual está compuesto por cinco características y dieciséis subcaracterísticas. Para asignar los pesos a las subcaracterísticas utilizan Analytic Hierarchy Process (AHP), en español proceso analítico jerárquico."

Techio & Misaghi (2015) proponen un modelo basado en los estándares de interoperabilidad, gestión de riesgos y auditoría de servicios TI. La organización del modelo consiste en cuatro dimensiones: aspectos técnicos, necesidades de los usuarios, mejores prácticas y viabilidad económica. Cada dimensión está formada por criterios y métricas. Entre los criterios evaluados se encuentran: flexibilidad, funcionalidad, acuerdos de nivel de servicio, retorno de la inversión, entre otros. Cada criterio tiene asociado una métrica, algunas de las métricas relacionadas en el estudio son: capacidad de ajustes, adaptación a las necesidades del usuario, política de prevención y corrección de errores, costos iniciales más costos directos e indirectos, entre otros.

Tan, Niu, Liu, & Yang (2016) proponen un modelo basado en la evaluación del cumplimiento de los acuerdos de nivel de servicio, calculando la confianza en el servicio. Este modelo soporta la trazabilidad del comportamiento de un proveedor de servicio. Los parámetros utilizados en este modelo para medir la calidad del servicio son: disponibilidad, confiabilidad, integridad, promedio del grado cumplimiento, similitud y valor integrado. La validación del modelo se realiza a través de simulaciones.

Shin & Huh (2016) definen MCSQAM como un modelo de evaluación de servicios móviles en la nube que se basa en el estándar ISO/IEC 9126. Los atributos que analizan para el servicio móvil en la nube son: funcionalidad, eficiencia, usabilidad y confiabilidad. Para la verificación del modelo propusieron un proceso de jerarquía analítica capaz de evaluar los servicios móviles según los requisitos de calidad.

Kara, Lamouchi, & Ramdane-Cherif (2017) proponen un modelo de datos general que contiene otros modelos de calidad que se conectan por medio de relaciones de equivalencia, que luego son programadas por medio de algoritmos que utilizan lógica

difusa. Las relaciones de equivalencia definidas son: igual nombre - igual significado, igual nombre – diferente significado, igual nombre – significado similar, diferente nombre – igual significado y diferente nombre - significado similar. El aporte de esta propuesta es permitir manipular, explorar y medir cualquier criterio de diferentes modelos de calidad de datos.

Olokunde, Misra, & Adewumi (2017) plantean un modelo para medir la calidad de las plataformas como servicio (Platform as a Service - PaaS). Definen las características principales de PaaS tales como: interoperabilidad, usabilidad y portabilidad. En la organización del modelo identifican los atributos de calidad para cada característica y presentan las métricas para cada atributo. Para la validación, se evalúan tres casos de estudio usando el proceso de jerarquía analítica.

2.2. Índice de medición del servicio

Como se enunció en el planteamiento del problema parte de las oportunidades de explorar en computación en la nube es la ausencia de unificación en los acuerdos de servicios de los proveedores, por tanto, es relevante conocer el estado actual del servicio y quienes son los proveedores existentes.

Gartner, en su informe sobre computación en la nube de mayo de 2018 presentó la clasificación de líderes, visionarios, especialistas y rivales (Smith, Leong, & Bala, 2018). La Figura 4 presenta los datos reportados por Gartner para el cuadrante mágico IaaS.



Figura 4. Cuadrante mágico para Cloud Infrastructure as a Service, Worldwide
Fuente: (Smith et al., 2018)

Como se puede apreciar en la Figura 4, no se identifican visionarios ni rivales. La computación en la nube está gobernada por 3 de los gigantes informáticos: Amazon, Microsoft y Google.

Los proveedores de servicios en la nube definen acuerdos de niveles de servicio para sus clientes, sin embargo, muy pocas veces el cliente es capaz de identificar qué es lo que está adquiriendo con un servicio en la nube. Para abordar esta problemática, la Universidad Carnegie Mellon creó la iniciativa Cloud Services Measurement Initiative Consortium (CSMIC), en español Consorcio para la Medición de los Servicios en la Nube, con el fin de desarrollar un marco de trabajo denominado el Service Measurement Index (SMI), en español índice de medición de servicio.

SMI define características, atributos y medidas para que un cliente pueda seleccionar un proveedor de servicios en la nube dependiendo de los servicios que ofrece. Se define como una estructura jerárquica de 7 categorías, cada una de ellas contiene atributos relacionados (CSMIC, 2014b). El objeto de este marco es suministrar medidas en cada una de las categorías para obtener valores de comparación de cada proveedor.

“Todo el trabajo del CSMIC en el SMI fue suspendido en julio de 2014” (CSMIC, 2014a), sin embargo, este marco de trabajo es una línea de base que ha sido tomada como referente por otros estudios en el área de calidad en computación en la nube (Saravanan, Aramudhan, Sundara Pandiyan, & Avudaiappan, 2018), (Ben-Abdallah, Boukadi, & Hammami, 2018), (Yadav & Goraya, 2018), (Somu, Kirthivasan, & V.S., 2017), (RĂDULESCU & RĂDULESCU, 2017), (Mohan & Aramudhan, 2017), (Totiya & Senivongse, 2017), (Tripathi, Pathak, & Vidyarthi, 2017), (Pintos, Castillo, & Lopez-Pires, 2016), (Aruna & Aramudhan, 2016), (Ruby, Aisha, & Subash, 2016), (M. Z. Khan & Qamar, 2015), (Ricardo Colomo-Palacios & Rodríguez, 2014), (Costa, Santos, & Da Silva, 2013), (Siegel & Perdue, 2012).

2.3. Normatividad relacionada con calidad de software

Para la construcción de la taxonomía propuesta en el presente estudio, se requiere el análisis de la normatividad existente sobre calidad en computación en la nube, esta sección describe los organismos y estándares que orientan la calidad en computación en la nube.

2.3.1. Organismos de normalización

En la Tabla 4 se presenta un listado con los organismos internacionales de normalización relacionados con la ingeniería incluida la ingeniería de software y afines.

Tabla 4. Organismos internacionales de normalización

Acrónimo	Nombre
ISO	Organización Internacional de Estandarización (ISO, 2018)
IEC	Comisión Electrotécnica Internacional (IEC, 2018)
ITU	Unión Internacional de Telecomunicaciones (ITU, 2018)
IEEE	Instituto de Ingeniería Eléctrica y Electrónica (IEE, 2018)

Los organismos que guardan relación con la calidad de software son IEEE e ISO. A continuación, se presentan los estándares descritos por estas organizaciones en el área de estudio del presente trabajo. Aunque IEC ha propuesto normas propias para la calidad de software, éstas no han sido adaptadas en su totalidad por ISO, incluso han retirado normas como la IEC PAS 62814:2012 - Dependability of software products containing reusable components - Guidance for functionality and tests (IEC, 2012), que aborda la confiabilidad de los productos software con componentes reutilizables. ISO e IEC han aunado esfuerzos para producir las normas actuales que sirven como referencia para la evaluación de la calidad de los productos software.

2.3.1.1. IEEE - Institute of Electrical and Electronics Engineers

IEEE es la organización profesional técnica más grande del mundo dedicada al avance de la tecnología para el beneficio de la humanidad (IEEE, 2018). Concentra sus esfuerzos en la estandarización para el desarrollo de las áreas relacionadas con las ingenierías y las nuevas tecnologías.

IEEE ha generado 8 estándares (Std) relacionados con la calidad del software y en conjunto con la ISO e IEC acordaron un estándar internacional de pruebas de software. En la Tabla 5 se presenta el número del estándar y su descripción.

Tabla 5. Estándares relacionados con la calidad del software

Nro.	Estándar	Descripción
1	IEEE Std 730-2014	Requerimientos para iniciar, planificar, controlar y ejecutar los procesos de calidad del software de un proyecto de desarrollo o mantenimiento (IEEE, 2014).
2	IEEE Std 982.1-2005	Diccionario estándar de medidas para construir software confiable (IEEE, 2006).
3	IEEE Std 1012-2016	Verificación y validación de sistema, software y hardware. Se utiliza para determinar si el producto cumple con los requisitos y si satisface las necesidades del usuario (IEEE, 2017).
4	IEEE Std 1028-2008	Revisión y auditoría de software, proporciona requisitos mínimos aceptables para realizar revisiones sistemáticas al software (IEEE, 2008).
5	IEEE Std 1044-2009	Plantea un enfoque para la clasificación de anomalías de software, independientemente de cuándo se originan o se

Nro.	Estándar	Descripción
		encuentran dentro del proyecto, producto o ciclo de vida del sistema (IEEE, 2010).
6	IEEE Std 1061-2005	Define una metodología para conformar requisitos de calidad, medir y validar métricas de calidad del proceso y producto software (IEEE, 1998).
7	IEEE Std 1228-2010	Se aplica a los planes de seguridad del software utilizados para el desarrollo, adquisición, mantenimiento y retiro de software (IEEE, 1994)
8	ISO/IEC/IEEE 29119	Conjunto de estándares reconocidos internacionalmente por las organizaciones ISO, IEC e IEEE para pruebas de software. Se pueden utilizar dentro de cualquier ciclo de vida u organización de desarrollo de software (ISO/IEC/IEEE, 2013).

2.3.1.2. ISO - International Organization for Standardization

ISO reúne a expertos para compartir conocimientos y desarrollar normas internacionales voluntarias, basadas en el consenso, que sean relevantes para el mercado y que respalden la innovación y brinden soluciones a los desafíos mundiales (ISO, 2018).

El comité de ingeniería de software y sistemas de la ISO con el apoyo de la IEC, desarrolló una serie de normas basadas en los estándares ISO/IEC 9126 e ISO/IEC 14598, estas normas llevan por nombre ISO/IEC 25000:2014, más conocidas como SQuaRE por las siglas en inglés Systems and software Quality Requirements and Evaluation. El objetivo principal de este conjunto de normas es la orientación del desarrollo de software con la definición y evaluación de requisitos de calidad (ISO/IEC, 2014). La Figura 5 presenta la organización por niveles de SQuaRE.

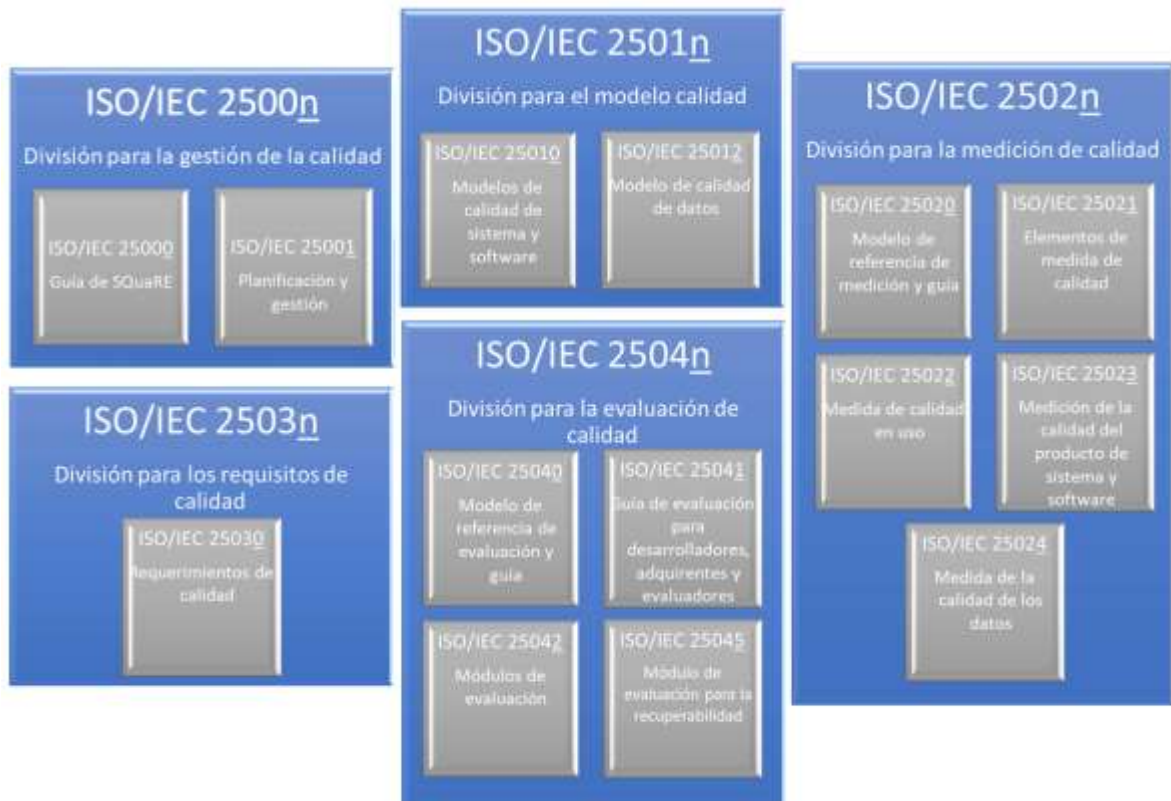


Figura 5. Niveles de SQuaRE.

La calidad en desarrollo software incluye tanto el proceso como el producto. SQuaRE está orientado al producto y la calidad se analiza desde tres vistas (ISO/IEC, 2014):

- Vista interna: Hace referencia al producto en desarrollo, analiza características como tamaño, modularidad, complejidad, entre otras.
- Vista externa: Hace referencia al producto en producción, analiza características como seguridad, rendimiento, escalabilidad, entre otras.
- Vista en uso: Hace referencia a la calidad en el uso del software, analiza características como productividad, efectividad, entre otras.

SQuaRE no define niveles de calidad óptimos para un desarrollo software, sin embargo, propone un conjunto de características que agrupan atributos de calidad (también llamados subcaracterísticas), los cuales pueden ser utilizados para evaluar el correcto funcionamiento del software. Un aspecto importante de SQuaRE, es que puede ser utilizado por empresas de software pequeñas o grandes.

2.3.2. Modelo de calidad seleccionado en este estudio

Los modelos de calidad definen una estructura de criterios a ser evaluados, los cuales están relacionados con los requerimientos iniciales. Los modelos son buenas prácticas para la construcción de software. La Figura 6 presenta la estructura de los modelos de calidad de software.



Figura 6. Estructura general de un modelo de calidad software.

Adaptada de “Modelos de calidad del software, un estado del arte”. M. Callejas-Cuervo, A. Alarcón-Aldana, A. Álvarez-Carreño. Entramado. p. 238. (Callejas-Cuervo, Alarcón-Aldana, & Álvarez-Carreño, 2017)

El enfoque de calidad de este estudio está orientado a la calidad del producto, por lo tanto, las normas de referencia deben tener el mismo enfoque. El IEEE Std 1061-2005 (IEEE, 1998), se basa en un modelo de calidad a la medida que no establece características fijas y tampoco métricas particulares, esto permite un nivel alto de adaptabilidad a cualquier proyecto. No obstante, la implementación de este enfoque requiere un esfuerzo inicial considerable y los resultados de su aplicación en diferentes contextos no son comparables.

Por su parte, las normas de ISO/IEC 2501n, agrupación denominada división para el modelo de calidad, utiliza un modelo mixto, el cual define características y atributos de calidad (subcaracterísticas) organizados en una jerarquía multinivel. El enfoque de este tipo de modelos radica en la definición de factores de calidad abstractos que puedan ser reutilizados en múltiples proyectos software. Por estas razones se analizarán en los siguientes apartados los componentes ISO/IEC 25010 (ISO/IEC, 2011) e ISO/IEC 25012 (ISO/IEC, 2008) que hacen parte de la división ISO/IEC 2501n.

2.3.3. Norma ISO/IEC 25010

Esta norma define el modelo de calidad del producto y del uso del software. El modelo de calidad de producto está dividido en ocho características, cada una de ellas tiene subcaracterísticas o atributos de calidad. Incluyen las características de seguridad y compatibilidad no incluidas en la norma ISO/IEC 9126 (P. Miguel, Mauricio, & Rodríguez, 2014). La Figura 7 presenta los atributos de calidad definidos en el modelo de calidad del producto.



Figura 7. Calidad del producto norma ISO/IEC 25010

El modelo de calidad de uso está relacionado con la aceptación que tiene el software sobre los interesados. Incluye tanto la calidad del software, como el hardware y el sistema sobre el cual está desplegado el producto final. Este modelo tiene cinco características y sus respectivos atributos de calidad. Ver Figura 8.



Figura 8. Calidad en uso norma ISO/IEC 25010.

2.3.4. Norma ISO/IEC 25012

Esta norma define el modelo para la calidad de los datos, entendiendo por datos, toda información almacenada de forma estructurada en un sistema de información. El modelo de calidad de los datos provee 15 características organizadas en dos categorías. La categoría calidad de datos inherente, agrupa las características que tienen la capacidad de satisfacer requerimientos propios de los datos, cuando son utilizados. Por ejemplo, relaciones entre los datos, metadatos, dominios y restricciones. La categoría calidad de

datos dependiente del sistema, agrupa las características que tienen la capacidad de preservar y alcanzar un nivel de calidad cuando los datos son utilizados, dependen del hardware y software que se utiliza para procesar los datos. La Tabla 6 presenta las dos categorías que conforman el modelo de calidad de los datos.

Tabla 6. Calidad de los datos

Características de calidad	Inherente	Dependiente del sistema
Precisión (Precision)	x	
Complejidad (Completeness)	x	
Consistencia (Consistency)	x	
Credibilidad (Credibility)	x	
Actualidad (Currentness)	x	
Accesibilidad (Accessibility)	x	x
Conformidad (Compliance)	x	x
Confidencialidad (Confidentiality)	x	x
Eficiencia (Efficiency)	x	x
Exactitud (Accuracy)	x	x
Trazabilidad (Traceability)	x	x
Entendibilidad (Understandability)	x	x
Disponibilidad (Availability)		x
Portabilidad (Portability)		x
Recuperabilidad (Recoverability)		x

Fuente: Norma ISO/IEC 25012. (ISO/IEC, 2008)

La Tabla 7 presenta un resumen de los hallazgos encontrados en esta sección.

Tabla 7. Resumen hallazgos trabajos relacionados

Perspectiva: Taxonomía		
Tipo	Nombre	Breve descripción
Artículo	A Taxonomy and Survey of Cloud Computing Systems	Estudio sobre las diferentes características que ofrecen los proveedores de servicios de computación en la nube, tanto privados como de acceso abierto, ofreciendo como resultado el alcance de cada uno, el cual puede ser empleado en la toma de decisiones para la adquisición de estos servicios.
Artículo	Cloud computing services: taxonomy and comparison. Journal of Internet Services and Applications,	Describen los actuales servicios de computación en la nube a través de una taxonomía para su clasificación diseñada en forma de árbol. Los principales servicios que hacen parte del estudio son: IaaS, PaaS, y SaaS, desplegando las características de calidad y servicios específicas para cada uno.
Artículo	A taxonomy model for cloud computing services	Estudio sobre conceptos existentes y taxonomías y armoniza estos dos enfoques en un modelo de taxonomía extensible para los servicios de computación en la nube, basada en la taxonomía SPI (Software, Platform and Infrastructure) creada por el NIST, para crear una organización jerárquica que agrupa diferentes servicios según sus características.
Artículo	Towards a Taxonomy of Performance Evaluation of Commercial Cloud Services	Introducen una taxonomía para ayudar a perfilar y estandarizar los detalles de la evaluación del desempeño de los servicios comerciales sobre computación en la nube, se concentran en características como comunicación, procesamiento, memoria, almacenamiento, velocidad de transacción, disponibilidad, latencia, confiabilidad, ancho de banda, escalabilidad y variabilidad.
Artículo	A Survey of Cloud Computing Taxonomies: Rationale and Overview	Proponen una comparación de taxonomías para computación en la nube para apoyar a investigadores y profesionales de la academia y la industria con la organización de conceptos y terminología sobre computación en la nube.
Artículo	Resource management in cloud computing: Taxonomy, prospects, and challenges.	Presentan una revisión exhaustiva de las técnicas de gestión de recursos (Resource Management -RM-) y elaboran una taxonomía basada en características como conciencia de consumo energético, acuerdos de niveles de servicio, orientación al mercado, balanceo, carga de red, nube híbrida y computación en la nube para móviles.

Perspectiva: Taxonomía		
Tipo	Nombre	Breve descripción
Reporte	NIST Cloud Computing Reference Architecture	Referente a nivel internacional en computación en la nube
Tesis doctoral	Modelo de Interoperabilidad para Plataformas de Cloud Computing basado en Tecnologías del Conocimiento	Conceptos más relevantes y emergentes en la actualidad como Cloud Computing, SaaS, tecnologías semánticas, modelado de procesos de negocio, interoperabilidad entre sistemas, entre otros, como base para definir un estándar o lenguaje común que permita la interoperabilidad entre aplicaciones dentro de un entorno de Cloud Computing.
Tesis doctoral	A semantic based policy management framework for cloud computing environments	Propuso un marco de gestión de políticas basado en la semántica para entornos de computación en la nube que permite a los usuarios especificar políticas de acceso que se aplican a sus recursos utilizando un marco unificado sin importar dónde se almacenen los recursos.

Perspectiva: Modelos de Calidad		
Tipo	Nombre	Breve descripción
Artículo	A Quality Model for Evaluating Software-as-a-Service in Cloud Computing.	Proponen un modelo para evaluar la calidad de SaaS. El modelo define cinco atributos de calidad: reusabilidad, disponibilidad, escalabilidad, confiabilidad y eficiencia.
Artículo	Integrated QoS utility-based model for cloud computing service provider selection.	Proponen un modelo matemático con el fin de optimizar el proceso de selección de los proveedores de servicios en la nube, con base en las garantías de la calidad del servicio. Los atributos seleccionados para este modelo son: tiempo de respuesta, rendimiento y tiempo disponible.
Artículo	Quality Model for Evaluating SaaS Service	Plantean un modelo de calidad que mide la seguridad, calidad del servicio y calidad de software del servicio SaaS. Esta evaluación es realizada desde tres perspectivas: de plataforma, del proveedor y del cliente.
Artículo	Cloudqual: A quality model for cloud services	Plantean un modelo de calidad para servicios en la nube denominado CLOUDQUAL. Este modelo contiene seis dimensiones de calidad: usabilidad, disponibilidad, confiabilidad, capacidad de respuesta, seguridad y elasticidad.
Artículo	A novel trust model for selection of Cloud Service Provider	Proponen un modelo orientado hacia la selección de servicios de proveedores eficientes y confiables, con base en el análisis de los datos recolectados por las

Perspectiva: Modelos de Calidad		
Tipo	Nombre	Breve descripción
		autoridades regulatorias.
Artículo	Quality Model of Cloud Service	Proponen un modelo de calidad que abarca características TI y servicios en la nube. El modelo de calidad se construyó basado en seis características: usabilidad, seguridad, confiabilidad, tangibilidad, capacidad de respuesta y empatía.
Artículo	Quality Model of Maintenance Service for Cloud Computing	Plantean dos modelos para la gestión de la calidad: 1) el primer modelo está orientado al mantenimiento del servicio desde dos perspectivas cliente y proveedor. 2) el segundo modelo para la evaluación del modelo de calidad, el cual está compuesto por cinco características y dieciséis subcaracterísticas.
Artículo	EMSCLOUD - An evaluative model of cloud services cloud service management	Proponen un modelo basado en los estándares de interoperabilidad, gestión de riesgos y auditoría de servicios TI. La organización del modelo consiste en cuatro dimensiones: aspectos técnicos, necesidades de los usuarios, mejores prácticas y viabilidad económica.
Artículo	A novel trust model based on SLA and behavior evaluation for clouds.	Proponen un modelo basado en la evaluación del cumplimiento de los acuerdos de nivel de servicio, calculando la confianza en el servicio.
Artículo	MCSQAM: Service Quality Assessment Model in Mobile Cloud Services Environment	Definen MCSQAM como un modelo de evaluación de servicios móviles en la nube que se basa en el estándar ISO/IEC 9126.
Artículo	Semantically equivalent model for quality evaluation.	Proponen un modelo de datos general que contiene otros modelos de calidad que se conectan por medio de relaciones de equivalencia, que luego son programadas por medio de algoritmos que utilizan lógica difusa.
Artículo	Quality Model for Evaluating Platform as a Service in Cloud Computing	Plantean un modelo para medir la calidad de las plataformas como servicio (Platform as a Service - PaaS). Definen las características principales de PaaS tales como: interoperabilidad, usabilidad y portabilidad.

Perspectiva: Índice de medición del servicio		
Tipo	Nombre	Breve descripción
Reporte	Magic Quadrant for Cloud Infrastructure as a Service, Worldwide	La computación en la nube está gobernada por 3 de los gigantes informáticos: Amazon, Microsoft y Google.
Informe	Service Measurement Framework Version 2.1 Index	La Universidad Carnegie Mellon creó la iniciativa Cloud Services Measurement Initiative Consortium (CSMIC), en español Consorcio para la Medición de los Servicios en la Nube, con el fin de desarrollar un marco de trabajo denominado el Service Measurement Index (SMI), en español índice de medición de servicio.
Perspectiva: Normatividad relacionada con calidad de software		
Tipo	Nombre	Breve descripción
Guía	Organismos de normalización	ISO Organización Internacional de Estandarización IEC Comisión Electrotécnica Internacional ITU Unión Internacional de Telecomunicaciones IEEE Instituto de Ingeniería Eléctrica y Electrónica
Estándar de calidad	de Estándar IEEE	IEEE Std 730-2014 IEEE Std 982.1-2005 IEEE Std 1012-2016 IEEE Std 1028-2008 IEEE Std 1044-2009 IEEE Std 1061-2005 IEEE Std 1228-2010 ISO/IEC/IEEE 29119
Estándar de calidad	de Estándar ISO	ISO/IEC 25000:2014 SQuaRE

3. VOCABULARIOS CONTROLADOS

3.1. Concepto

De acuerdo con NISO (ANSI/NISO, 2005), un vocabulario controlado es una lista de términos que se han enumerado explícitamente. Esta lista está controlada y disponible por una autoridad de registro de vocabulario controlado. Todos los términos en un vocabulario controlado deben tener una definición inequívoca y no redundante. Garshol (2004) lo define como una lista cerrada de temas nombrados, que pueden ser utilizados para la clasificación. Los componentes de un vocabulario controlado se conocen generalmente como términos, donde un término es un nombre particular para un concepto particular.

En el contexto de los sistemas de organización del conocimiento, los vocabularios controlados son una pieza fundamental para la construcción de taxonomías, tesauros y ontologías.

Ahora bien, un vocabulario controlado debe cumplir como mínimo con las siguientes dos reglas (ANSI/NISO, 2005):

1. Si el mismo término se utiliza comúnmente para significar conceptos diferentes, entonces su nombre está explícitamente calificado para resolver esta ambigüedad.
2. Si se utilizan términos múltiples para significar lo mismo, uno de los términos se identifica como el término preferido en el vocabulario controlado y los otros términos se enumeran como sinónimos o alias.

A través de un vocabulario controlado se busca asegurar consistencia en la aplicación del lenguaje (Hedden, 2010). Se dice que es controlado porque únicamente los términos de la lista deberían ser utilizados para referirse a los conceptos del área temática cubierta por el vocabulario.

3.2. Sistemas de organización del conocimiento

Los sistemas de organización del conocimiento contemplan diferentes propósitos y diferentes complejidades estructurales. En la Tabla 8 se presentan en orden de complejidad descendente junto con su definición.

Tabla 8. Sistemas de organización del conocimiento

Tipo	Definición
Ontología	Diseños de estructuras funcionales, que contienen entidades o elementos que se relacionan entre sí, para llevar a cabo determinados propósitos o para cumplir con ciertos objetivos.
Tesauros	Sistemas totalmente estructurados, que se integran con términos que guardan entre sí relaciones semánticas y funcionales, que tienen por objeto proporcionar

Tipo	Definición
	un instrumento idóneo para el almacenamiento y la recuperación de la información en áreas especializadas.
Taxonomías	Términos conectados mediante algún modelo estructural (jerárquico, arbóreo, facetado) y especialmente orientado a los sistemas de navegación, organización y búsqueda de contenido. Una taxonomía es una colección de términos de vocabularios controlados organizados en una estructura jerárquica (ANSI/NISO, 2005).
Vocabulario controlado	Es un conjunto de términos normalizados dentro de un área del conocimiento que se caracteriza por identificar descriptores que definen de manera única un concepto.
Anillos de sinónimos	Conjuntos de términos que son considerados equivalentes para los propósitos de la recuperación de información, y por ese motivo se ponen a disposición de los usuarios, con la finalidad de orientar las búsquedas a texto libre. En consecuencia, están destinados antes a los procesos de recuperación temática de información, que a la indización. No establecen ninguna jerarquía ni discriminación entre los sinónimos o cuasi sinónimos.
Listas	En su forma más simple un vocabulario controlado consta únicamente de una lista de términos (cada uno asociado a un concepto distinto). En las listas sólo interesa eliminar la ambigüedad, para esto se define un único término para referirse a un concepto.

Fuente: Tomado de "Sistemas de organización del conocimiento: Una tipología actualizada" (Barité-Roqueta, 2011)

3.3. Principios de los vocabularios controlados

Existen un conjunto de razones o principios que justifican la construcción de un vocabulario controlado. Ver Tabla 9.

Tabla 9. Principios de los vocabularios controlados

Principio	Descripción
Eliminación de ambigüedad	En el lenguaje natural ocurre ambigüedad cuando una palabra o frase tiene más de un significado. Un vocabulario controlado debe compensar los problemas de ambigüedad asegurando que cada término tenga un único significado.
Control de sinónimos	La sinonimia ocurre cuando un concepto puede ser representado por dos o más términos diferentes. Un vocabulario controlado debe compensar los problemas causados por la sinonimia asegurando que cada concepto sea representado por un término preferido. Los términos equivalentes para un mismo concepto serán registrados en el vocabulario indicando que debe utilizarse el término preferido para representarlo.
Definición de relaciones entre los términos	Un vocabulario controlado debe modelar las relaciones entre los términos que sean necesarias. Es posible modelar relaciones de equivalencia, relaciones jerárquicas (indicando que un concepto es más específico que su concepto padre) y relaciones asociativas (indicando otro tipo de

Principio	Descripción
Prueba y validación de términos	relaciones entre dos conceptos). El proceso de construcción de un vocabulario controlado involucra consultar varias fuentes de palabras y frases, así como criterios basados en: el lenguaje natural utilizado para describir objetos de contenido (garantía literaria), el lenguaje de los usuarios (garantía de los usuarios) y las necesidades de la organización (garantía organizacional).

Fuente: Tomado de "Hacia una taxonomía sobre educación de ingeniería del software (Pizard, 2016)

3.4. Garantía literaria

La construcción de un vocabulario controlado debe apoyarse sobre elementos que le den validez. (Barité, 2009) expone el concepto de garantía literaria, sustentada en la idea central de que la literatura de un dominio debe ser la fuente de extracción y validación de la terminología que se ha de incorporar en un sistema de clasificación. Constituye una formulación simple y de fácil comprensión, apoyada en el sentido común. Ofrece una salida metodológica para la difícil tarea de representar el conocimiento de tal modo que sea accesible para usuarios de todos los niveles de instrucción y con intereses y necesidades muy variadas de información.

De igual forma, se recomienda utilizar la garantía literaria por su beneficio pragmático, debido a que la función típica de la clasificación es la organización lógica de un conjunto de documentos. Al utilizar los mismos documentos para crear el sistema de clasificación es natural que ese sistema los permita clasificar. A través de la garantía literaria, entre otros mecanismos, se facilitan los procesos de desambiguación y relación entre términos.

Por otro lado la norma ANSI/NISO Z-39.19 (ANSI/NISO, 2005) define la garantía literaria como la justificación para la representación de un concepto en un lenguaje de indexación o para la selección de un término preferido debido a su frecuente presencia en la literatura.

La construcción de un vocabulario controlado debe contar con mecanismos que den validez al resultado obtenido, por cuanto la garantía literaria apoya la selección de los conceptos que hagan parte de dicho vocabulario.

4. PROTOCOLO PARA LA CONSTRUCCIÓN DEL VOCABULARIO

4.1. Necesidad del protocolo

El protocolo propuesto en esta tesis define los lineamientos para realizar una búsqueda sistemática que permita identificar, evaluar e interpretar las investigaciones con resultados relacionados con atributos de calidad en computación en la nube. El protocolo toma como referencia las sugerencias propuestas por Kitchenham (2007) para realizar revisiones sistemáticas de literatura. La definición del protocolo permite que otros investigadores puedan verificar el rigor metodológico y la integridad del trabajo realizado.

4.2. Planificación de la revisión sistemática

4.2.1. Necesidad de la revisión

Sobre la temática de computación en la nube, existen trabajos de alto impacto que sustentan la importancia de las características de calidad en computación en la nube (Rimal et al., 2009), (Hoefler y Karagiannis, 2010, 2011), (Gonzalez et al., 2011), (Z. Li et al., 2012), (Polash et al., 2014), y (Mustafa et al., 2015). En estos trabajos los autores abordaron las temáticas de calidad en computación en la nube desde el punto de vista de taxonomías. Sin embargo, ninguno de estos autores profundizó en las características de calidad.

Por su parte, la Universidad Carnegie Mellon aportó el índice de medición del servicio (CSMIC, 2014b), que estableció un conjunto de indicadores para definir un método que permitiera seleccionar proveedores.

En la revisión de la literatura se analizaron estudios que proponen modelos de calidad, sobresalen los trabajos realizados por J. Y. Lee et al., (2009), Salama et al., (2012), Wen & Dong (2013), X. Zheng et al., (2014), M. K. Naseer et al., (2014), Zhou et al., (2015), Z. Wang et al., (2015), Techio & Misaghi (2015), Z. Tan et al., (2016), Shin & Huh (2016), Kara et al., (2017) y Olokunde et al., (2017). No obstante, todos estos trabajos abordan las características de calidad desde perspectivas particulares.

Por lo anterior, fue necesario realizar una revisión sistemática de literatura sobre características de calidad en computación en la nube, para tener una base de conocimiento que permitiera organizar y relacionar las características de calidad en la nube desde una perspectiva global.

4.2.2. Identificación de la investigación

Petticrew & Roberts (2005) sugieren utilizar los criterios PICOC (por sus siglas en inglés) Población, Intervención, Comparación, Resultado y Contexto, para formular las preguntas

de investigación. En consecuencia, se presenta en la Tabla 10 el análisis de cada uno de los criterios.

Tabla 10. Criterios PICOC para la definición de las preguntas de investigación

Criterio	Descripción
Población	Trabajos de investigación publicados en fuentes de alto impacto sobre temas de calidad en computación en la nube.
Intervención	Clasificación de atributos de calidad y relación entre ellos.
Comparación	Grupos de atributos y posibles clasificaciones encontradas.
Resultado	Base conceptual de atributos de calidad utilizados en el despliegue de aplicaciones en la nube y sus posibles relaciones conceptuales, sirve como insumo para la construcción de la taxonomía.
Contexto	Revisión de literatura sobre el dominio de atributos de calidad en aplicaciones diseñadas para ser desplegadas en la nube.

Como se puede apreciar en la Tabla 10, el objetivo de esta revisión sistemática es identificar los trabajos de investigación que abordan los temas de calidad en computación en la nube desde la perspectiva de los atributos de calidad. Por lo anteriormente expuesto, se proponen las siguientes preguntas de investigación:

1. ¿Existen estudios que permitan conocer el estado actual de los atributos de calidad en computación en la nube?
2. ¿Qué atributos de calidad se utilizan en las aplicaciones al ser desplegadas en la nube?
3. ¿Existe alguna clasificación para los atributos de calidad en aplicaciones a ser desplegadas en la nube?

4.3. Desarrollo del protocolo

4.3.1. Repositorios

La selección de los repositorios está soportada en el trabajo realizado por Brereton, Kitchenham, Budgen, Turner, & Khalil (2007). En la Tabla 11 se relacionan los repositorios utilizados para la búsqueda de información.

Tabla 11. Repositorios digitales seleccionados

Nombre del repositorio	URL
IEEE Explore	https://ieeexplore.ieee.org/Xplore/home.jsp
SCOPUS	https://www.scopus.com/home.uri
ACM Digital Library	https://dl.acm.org/
ScienceDirect	https://www.sciencedirect.com/

4.3.2. Palabras claves

Teniendo en cuenta las preguntas de investigación, se definieron las palabras claves a utilizar en la búsqueda sistemática de artículos. La Tabla 12 relaciona las palabras claves.

Tabla 12. Palabras claves

Palabras claves
Cloud
Computing
Quality
Attributes
Cloud computing
Quality attributes

De igual forma se definieron filtros adicionales con el fin de refinar las búsquedas al área de conocimiento específico del presente trabajo. La Tabla 13 presenta los filtros aplicados.

Tabla 13. Filtros adicionales para refinar las búsquedas

Filtros adicionales
Software engineering
Computer science
Computer engineering

4.3.3. Ecuación de búsqueda

En el proceso de definición y calibración de la ecuación se utilizaron combinaciones como: Quality of Experience (QoE), Quality of Service (QoS), quality cloud computing, quality model, entre otros. Sin embargo, se evidenció la exclusión de trabajos de investigación relevantes. Por lo tanto, se utilizó el concepto de amplia cobertura, planteado por Kitchenham (2007).

Encontrar la ecuación de búsqueda definitiva es una tarea exhaustiva, requiere dos elementos fundamentales: 1) conocimiento de la temática en cuestión y 2) heurística para encontrar la mejor alternativa. Teniendo en cuenta lo enunciado anteriormente, las preguntas de investigación definidas en este trabajo y las palabras claves identificadas, se estableció la siguiente ecuación de búsqueda, independiente de los repositorios.

(Cloud Computing) AND ((quality OR attribute*))

En la Figura 9 se puede apreciar la relación existente entre las palabras claves y el modelo PICOC propuesto por Petticrew & Roberts (2005) para construir las preguntas de investigación.



Figura 9. Relación PICOC y la ecuación de búsqueda.

4.3.4. Ejecución de la ecuación de búsqueda

La ejecución de la ecuación de búsqueda se llevó a cabo en tres momentos: 16 de octubre de 2017, 12 de febrero de 2018 y 18 de junio de 2018. La ventana de observación para todos los repositorios incluye los trabajos publicados desde el año 2006 en adelante. El año 2006 es un hito en computación en la nube por dos razones: 1) aparece en escena la plataforma Google Docs y 2) se consolida Amazon Web Services a través de la plataforma Amazon EC2 (Elastic Compute Cloud por sus siglas en inglés). Posterior a la ejecución de la ecuación en cada repositorio, se eliminaron los trabajos duplicados. La Tabla 14, presenta la información de los trabajos encontrados.

Tabla 14. Filtro por trabajos duplicados

Repositorio	Encontrados	Duplicados	Para revisión
IEEE Explore	2696	---	2696
SCOPUS	5543	2208 IEEE	3335
ACM Digital Library	686	56 IEEE 248 Scopus	382
ScienceDirect	136	---	136
Total			6549

La Tabla 15 presenta las ecuaciones personalizadas para cada repositorio y los resultados de las búsquedas en cada uno de los tres momentos.

Tabla 15. Ecuaciones de búsqueda personalizadas por repositorio

Repositorio	Ecuación	Fecha	Cantidad
IEEE Explore	(((("Document Title":Cloud Computing) OR "Abstract":Cloud Computing) AND ("Document Title":quality OR "Abstract":quality OR "Document Title":attributes OR "Abstract":attributes))) Filters Applied: 2006-2018	18-jun-2018	2696
		12-feb-2018	2521 (-6%)
		16-oct-2017	2350 (-13%)
SCOPUS	(TITLE (cloud AND computing) OR ABS (cloud AND computing)) AND (TITLE (quality) OR ABS (quality) OR TITLE (attributes) OR ABS (attributes)) AND (LIMIT-TO (PUBYEAR , 2019) OR LIMIT-TO (PUBYEAR , 2018) OR LIMIT-TO (PUBYEAR , 2017) OR LIMIT-TO (PUBYEAR , 2016) OR LIMIT-TO (PUBYEAR , 2015) OR LIMIT-TO (PUBYEAR , 2014) OR LIMIT-TO (PUBYEAR , 2013) OR LIMIT-TO (PUBYEAR , 2012) OR LIMIT-TO (PUBYEAR , 2011) OR LIMIT-TO (PUBYEAR , 2010) OR LIMIT-TO (PUBYEAR , 2009) OR LIMIT-TO (PUBYEAR , 2008) OR LIMIT-TO (PUBYEAR , 2007) OR LIMIT-TO (PUBYEAR , 2006)) AND (LIMIT-TO (SUBJAREA , "COMP") OR LIMIT-TO (SUBJAREA , "ENGI")) *	18-jun-2018	3335
		12-feb-2018	3206 (-4%)
		16-oct-2017	2968 (-11%)
ACM Digital Library	(acmdlTitle:(+Cloud +Computing) OR recordAbstract:(+Cloud +Computing)) AND (acmdlTitle:(+quality) OR acmdlTitle:(+attributes) OR recordAbstract:(+quality) OR recordAbstract:(+attributes)) Published since: 2006	18-jun-2018	382
		12-feb-2018	362 (-5%)
		16-oct-2017	342 (-10%)
ScienceDirect	(("Cloud Computing") AND ("quality attributes")) Refine your search Journals, all Sciences, 2006 to present	18-jun-2018	136
		12-feb-2018	121 (-11%)
		16-oct-2017	113 (-17%)

Un análisis cuantitativo de los resultados presentados en la Tabla 15 permite inferir que, a mayo de 2019, podrían existir un 13% más de publicaciones sobre la temática en estudio, asumiendo un crecimiento lineal. Este factor evidencia la oportunidad latente para continuar realizando trabajos relacionados con la temática de estudio.

4.3.5. Preselección de trabajos

4.3.5.1. *Procesamiento automático*

Una red bibliométrica está compuesta por nodos y bordes (Van Eck & Waltman, 2014). Para el presente estudio los nodos son los trabajos encontrados tras la ejecución de las cadenas de búsqueda en cada repositorio. Para cada nodo de la red se procesa el contenido del título y el resumen. Los bordes por su parte indica la relación y la fuerza que existe entre los nodos.

Para la preselección de los trabajos se utilizó una herramienta de visualización de panoramas científicos o redes bibliométricas denominada VOSviewer (Leiden University, 2018), esta herramienta utiliza la técnica de normalización por asociación de fuerza discutida por Eck & Waltman (2009). El objeto de la herramienta utilizada es la visualización bidimensional en mapas etiquetados, donde cada etiqueta tiene un tamaño proporcional a su peso. Las etiquetas representadas se agrupan en clústeres y van desde el color rojo hasta el color azul, reflejando la densidad de las relaciones entre cada etiqueta. Si existe una gran densidad, es decir, mayor concurrencia de descriptores, la etiqueta se representará con un color rojo, a medida que aumenta la dispersión de las etiquetas la representación se tornará de color azul. La Figura 10 presenta la visualización de la red bibliométrica para el presente estudio, teniendo en cuenta el tercer momento correspondiente al 18 de junio de 2018.

Van Eck & Waltman (2010) presentan en detalle la técnica de visualización de similitudes utilizada por la herramienta VOSviewer. La importancia del uso de esta herramienta radica en la asignación de los nodos y sus etiquetas a un conjunto de clústeres, los cuales permiten clasificar el objeto de estudio y de esta manera limitar la investigación. Conforme a lo propuesto por Van Eck & Waltman (2014), en este estudio se utilizó la metodología de conteo fraccional (Perianes-Rodriguez, Waltman, & Van Eck, 2016).

VOSviewer estableció cinco clústeres que agrupan 1663 descriptores. La Tabla 16 presenta la clasificación y el enfoque de los descriptores encontrados.

Tabla 16. Clústeres definidos por VOSviewer

Nro	Descriptores	Enfoque
Clúster 1 Rojo	825	Agrupó los descriptores de tipos de servicios, aplicaciones, disponibilidad, tipos de nubes, computación, datos, demanda, economía, computación de borde, eficiencia, elasticidad, evaluación, experimentos, fallas, calidad, rendimiento, infraestructura, computación móvil, redes, calidad del servicio, SLA, proveedores, entre otros.
Clúster 2 verde	255	Agrupó los descriptores de cifrado, atributos de cifrado, servidores, autenticación, autorización, confidencialidad, ataques, compartiendo datos, identidad, permisos, protección, seguridad de datos, acceso de usuarios, entre otros.
Clúster 3 azul	229	Agrupó los descriptores de proyectos, sistemas de información, credibilidad, decisiones, economía, salud, madurez, roles, perspectivas, entre otros.
Clúster 4 Amarillo	220	Agrupó los descriptores de tópicos, aprendizaje, teorías, contenidos, ingeniería, redes neuronales, detección, computación inteligente, fuzzy, ruido, métodos de predicción, señales, agua, redes inalámbricas, entre otros.
Clúster 5 Purpura	134	Agrupó los descriptores de negocios, marcos de trabajo conceptuales, documentación, ecosistemas, gobierno, salud, educación, planeación, enseñanza, virtualización, entre otros.

De la relación de los descriptores y la agrupación de los clústeres, se evidencia que los trabajos pertinentes con la investigación son los que se encuentran en el clúster número uno. La Tabla 17 presenta la cantidad de trabajos preseleccionados por cada uno de los repositorios consultados.

Tabla 17. Trabajos preseleccionados automáticamente

Repositorio	Trabajos para revisión	Preseleccionados
IEEE Explore	2696	1757
SCOPUS	3335	1987
ACM Digital Library	382	247
ScienceDirect	136	92
Total	6549	4083

La cantidad de trabajos preseleccionados de manera automática, corresponden al 62% del total de los estudios identificados para revisión. Este proceso automático descartó 2466 trabajos.

4.3.5.2. *Procesamiento manual*

Para seleccionar los estudios primarios que hacen parte del proceso de extracción de datos, se debían analizar 4083 trabajos, identificados en el procesamiento automático. No obstante, descartar 2466 trabajos fue considerado como una amenaza a la validez de la investigación. Para minimizar la amenaza a la validez de la investigación, se realizó el procesamiento manual a la totalidad de los trabajos identificados para revisión.

El protocolo para el procesamiento manual de los 6549 trabajos inició con el análisis de los metadatos: título, resumen y palabras claves. Este análisis permitió realizar un primer filtro para obtener trabajos que apoyaran el objetivo de esta investigación. En la Tabla 18 se definen los aspectos analizados en cada metadato.

Tabla 18. Criterios para la preselección de trabajos

Orden de lectura	de Metadato	Criterio
1	Título	El título debe enmarcar el direccionamiento del trabajo realizado, por ende, debe indicarle al revisor si incluye aspectos de calidad en computación en la nube.
2	Resumen	Trabajos que presenten aspectos de computación en la nube con respecto a: atributos, métricas y modelos de calidad.
3	Palabras claves	Si luego de la lectura del título y del resumen no se logra identificar el alcance del trabajo, se analizan las palabras claves. Se preseleccionarán los trabajos que incluyan mínimo dos de las siguientes palabras claves: calidad, atributos, métricas, modelos de calidad.

Los criterios de preselección se aplicaron a la totalidad de los trabajos identificados con la ecuación de búsqueda. La Tabla 19 presenta el resultado de la implementación de los criterios.

Tabla 19. Trabajos preseleccionados manualmente

Repositorio	Para revisión	Preseleccionados
IEEE Explore	2696	220
SCOPUS	3335	112
ACM Digital Library	382	37
ScienceDirect	136	20
		389

4.3.5.3. Comparación del procesamiento automático y manual

Una vez finalizada la fase de procesamiento manual, se realizó la comparación de los resultados entre el procesamiento automático y el manual. La Tabla 20 presenta la información de los trabajos preseleccionados en cada uno de estos procesos.

Tabla 20. Comparación procesos

Repositorio	No preseleccionados Automático	No preseleccionados Manual	Falso negativo	% Exclusión Automático
IEEE Explore	939	2476	38	4%
SCOPUS	1348	3223	19	1.4%
ACM Digital Library	135	345	3	2.2%
ScienceDirect	44	116	1	2.3%
Total	2466	6160	61	2.47%

La columna falso negativo de la Tabla 20 representa los trabajos que el procesamiento automático descartó, pero que el procesamiento manual si incluyó.

La columna porcentaje de exclusión automático se calcula al dividir los falsos negativos sobre los no preseleccionados automáticamente. El procesamiento manual incluyó 61 trabajos que fueron excluidos por el procesamiento automático, lo cual corresponde al 2.47% de los trabajos procesados.

4.3.6. Evaluación de la calidad de los trabajos preseleccionados

De acuerdo con Kitchenham (2007), una vez se define la muestra inicial de los trabajos, es necesario evaluar adecuadamente su relevancia, por tal razón, se definió el instrumento presentado en la Figura 11, el cual está basado en las preguntas de investigación que se obtuvieron al aplicar la metodología PICOC.

Preguntas	Identificadores de los trabajos	Condición inclusión	Repositorios seleccionados																							
			IEEE				SCOPUS				ACM				ScienceDirect											
1. ¿Es un trabajo de investigación? Aspectos por analizar: 1. Resultado de una investigación 2. No incluye casos de estudios limitados 3. Sólida fundamentación y referencias		AND	8	10	11	42	8	10	25	41	2	5	11	14	4	8	17	26
2. ¿Es un trabajo cuyo objetivo principal es el análisis de atributos y/o métricas de calidad? Aspectos por analizar: 1. Los atributos y/o métricas tienen su definición 2. Se evidencia un análisis sobre los atributos y/o métricas referenciados		AND																								

Figura 11. Instrumento para evaluar la calidad de los trabajos preseleccionados.

El instrumento presentado contiene dos preguntas con 5 aspectos a evaluar. La primera pregunta hace referencia a la calidad del trabajo en términos de investigación, alcance y

fundamentación. La segunda pregunta busca identificar los trabajos que aporten a la construcción de la taxonomía. Para que un trabajo sea seleccionado debe cumplir las dos preguntas en sus aspectos respectivos. La Tabla 21 presenta los resultados de la aplicación del instrumento a los 389 trabajos preseleccionados.

Tabla 21. Trabajos que aprobaron la evaluación de calidad

Repositorio	Preseleccionados	Seleccionados
IEEE Explore	220	70
SCOPUS	112	32
ACM Digital Library	37	5
ScienceDirect	20	12
Total	389	119

4.3.7. Línea de base para la extracción

El instrumento para apoyar la extracción de datos no tiene relevancia por sí solo, el conocimiento que permite alimentar la herramienta y posteriormente la dinámica que permite su mejora son fundamentales al momento de utilizar el instrumento. La estrategia de utilizar una línea de base adecuada para alimentar el instrumento permite tener una idea general de hacia dónde puede llegar la taxonomía. Por lo anterior, se tomaron como referencia dos líneas de base: 1. El marco de trabajo Service Measurement Index (SMI), desarrollado por el Carnegie Mellon (CSMIC, 2014b) y 2. El grupo de clústeres creado por la herramienta carrot2 con base en los trabajos que aprobaron la evaluación de calidad.

4.3.7.1. SMI

Inicialmente el instrumento se cargó con la información propuesta por el SMI, esta estrategia permitió que durante el proceso de extracción se pudieran ubicar temporalmente características y subcaracterísticas en grupos iniciales. A medida que se fueron procesando los trabajos, se fueron estableciendo las relaciones entre las características y subcaracterísticas encontradas. La Tabla 22 presenta las características propuestas por el SMI.

Tabla 22. Características propuestas por el SMI

Características	Subcaracterísticas
Accountability	1. Auditability, 2. Compliance, 3. Contracting experience. 4. Ease of doing Business, 5. Governance, 6. Ownership 7. Provider business stability, 8. Provider Certifications 9. Provider Contract/SLA Verification, 10. Provider Ethicality 11. Provider Personnel Requirements, 12. Provider Supply Chain 13. Provide Support, 14. Sustainability
Agility	1. Adaptability, 2. Elasticity, 3. Extensibility, 4. Flexibility 5. Portability, 6. Scalability
Assurance	1. Availability, 2. Maintainability, 3. Recoverability, 4. Reliability 5. Resiliency/Fault Tolerance, 6. Service stability, 7. Serviceability

Características	Subcaracterísticas
Financiamiento	1. Billing Process, 2. Cost, 3. Financial Agility, 4. Financial Structure
Seguridad y Privacidad	1. Access Control & Privilege Management, 2. Data Geographic/Political 3. Data Integrity, 4. Data Privacy & Data Loss 5. Physical & Environmental Security 6. Proactive Threat & Vulnerability Management, 6. Retention/Disposition 7. Security Management
Usabilidad	1. Accessibility, 2. Client Personnel Requirements, 3. Installability 4. Learnability, 5. Operability, 6. Transparency, 7. Understandability

Fuente: Best Cloud Provider Selection using Integrated ANP-DEMATEL and Prioritizing SMI Attributes. (Ghafori & Manouchehri Sarhadi, 2013)

4.3.7.2. Carrot2

Es un motor que permite agrupar resultados de búsquedas y los organiza por categorías temáticas. Esta herramienta ofrece actualmente dos algoritmos de agrupación: Lingo (Osiński, Stefanowski, & Weiss, 2004) y Suffix Tree Clustering (STC) (Jongkol Janruang, 2011). La diferencia entre los algoritmos radica en el principio de agrupamiento, lo que se traduce en calidad y rendimiento. Carrot2 ofrece componentes para obtener datos de Microsoft Bing o PubMed (Trueba-Gómez & Estrada-Lorenzo, 2010). Adicionalmente, provee otras fuentes para procesamiento de documentos como Lucene (Recio-García, Díaz-Agudo, & González-Calero, 2008), Apache Solr (Korkontzelos, Mu, & Ananiadou, 2012) o Elasticsearch (P. K. Singh, Suryawanshi, Gupta, & Saindane, 2016).

En esta investigación se utilizó el algoritmo Lingo, el cual identifica las etiquetas de cada clúster y luego asocia los documentos a cada una de las etiquetas. Lingo unido con Lucene permiten procesar documentos PDF sin necesidad de crear archivos XML para su procesamiento. Básicamente, Lingo crea una matriz de términos a partir de todos los índices de los documentos creados por Lucene, luego crea vectores base para hacer comparaciones y aproximaciones a la matriz. Finalmente, los vectores se convierten en las etiquetas que conforman los clústeres.

Lingo y STC pueden asociar un documento a dos o más clúster, lo cual contradice los principios para la construcción de taxonomías. Adicionalmente, Carrot2 necesita de una cadena de búsqueda para crear los clústeres. La herramienta por sí sola no funciona, requiere del conocimiento del experto en la temática de estudio. El investigador definió las cadenas de búsqueda teniendo como línea de base el SMI.

Para la construcción del clúster principal se utilizó la siguiente cadena de consulta:

***usability security privacy performance financial assurance agility
accountability quality attrib metric***

La consulta debe ser ejecutada sin incluir caracteres especiales o separadores en la expresión, si este es el caso, se generaría un clúster totalmente diferente que puede constituirse en una amenaza a la validez del estudio por considerarse un sesgo de información. La Figura 12 presenta el clúster de nivel 1 con 31 nodos generado por Carrot2. Los valores entre paréntesis en cada uno de los nodos corresponden a la cantidad de trabajos que Carrot2 asignó a ese nodo.

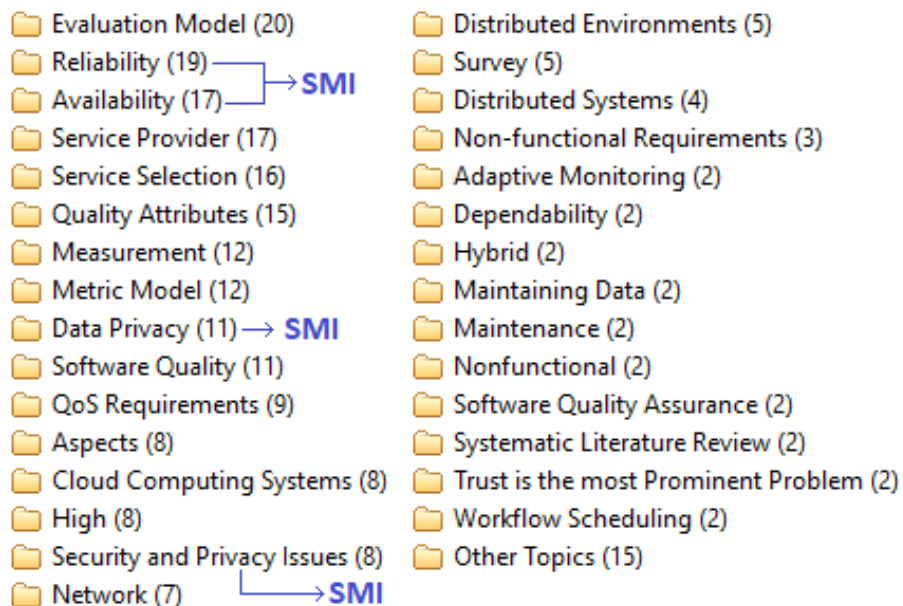


Figura 12. Clúster nivel 1 generado por Carrot2

Como se puede apreciar en la Figura 12, algunos de los nodos corresponden a las características propuestas por SMI. Para construir cada nodo interno, se debe tener una consulta independiente. Para el nodo “*Evaluation Model (20)*” se utilizó la siguiente cadena de búsqueda:

***security privacy usability metric quality interoperability usability
portability attribute performance accountability assurance agility cost
reliability flexibility service***

La Figura 13 presenta el clúster de nivel 2 generado para el nodo “*Evaluation Model (20)*”.

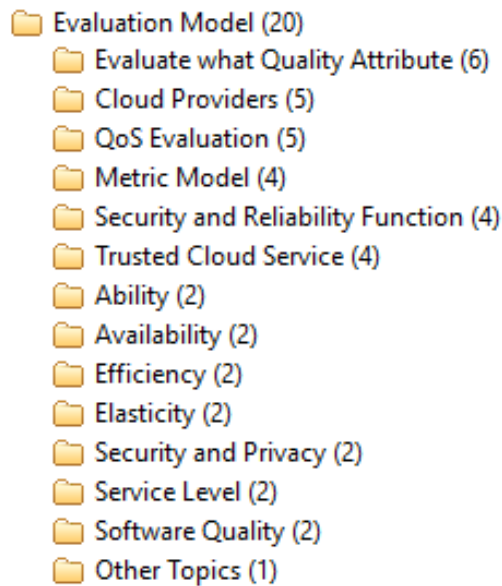


Figura 13. Clúster nivel 2 para nodo “Evaluation Model (20)” generado por Carrot2

Los artículos agrupados en el nodo “Evaluation Model (20)” se presentan en la Tabla 23.

Tabla 23. Artículos del nodo “Evaluation Model (20)”

Nro.	Nombre de artículo	Referencia
1.	An Approach to IdM with Privacy in the Cloud.	(Werner, Westphall, Weingartner, Geronimo, & Westphall, 2015)
2.	Quality Model for Evaluating Platform as a Service in Cloud Computing.	(Olokunde et al., 2017)
3.	A Quality Model for Evaluating Software-as-a-Service in Cloud Computing.	(J. Y. Lee et al., 2009)
4.	Towards Service Evaluation and Ranking Model for Cloud Infrastructure Selection.	(M. Z. Khan & Qamar, 2015)
5.	Quality evaluation and best service choice for cloud computing based on user preference and weights of attributes using the analytic network Process.	(Choi & Jeong, 2014)
6.	Cloud service evaluation method-based Multi-Criteria Decision-Making: A systematic literature review.	(Alabool, Kamil, Arshad, & Alarabiat, 2018)
7.	Designing an efficient methodology based on Entropy-TOPSIS for evaluating efficiency of cloud services	(R. R. Kumar & Kumar, 2017)
8.	The evaluation system for cloud service quality based on SERVQUAL	(Hu & Zhang, 2013)
9.	QoS evaluation for web services in cloud computing	(Nadanam & Rajmohan, 2012)

Nro.	Nombre de artículo	Referencia
10.	Software metric evaluation on cloud-based applications	(Phani Sheetal & Ravindranath, 2018)
11.	Research on the measurement and evaluation of trusted cloud service	(Ma, Jiang, Yang, Li, & Zhang, 2018)
12.	Modeling Correlation between QoS Attributes for Trust Computation in Cloud Computing Environments	(Mrabet, Saied, & Saidane, 2017b)
13.	A QoS Evaluation Model for Test-Bed in the Cloud Computing Environment	(Song et al., 2012)
14.	An Evaluation Model for Selecting Cloud Services from Commercially Available Cloud Providers	(Wagle, Guzek, Bouvry, & Bisdorff, 2016)
15.	A Stochastic Model to Investigate Data Center Performance and QoS in IaaS Cloud Computing Systems	(Bruneo, 2014)
16.	Semantically Equivalent Model for Quality Evaluation	(Kara et al., 2017)
17.	Optimal Negotiation of Service Level Agreements for Cloud-Based Services through Autonomous Agents	(Yaqub et al., 2014)
18.	An empirical study for evaluating the performance of multi-cloud APIs	(Ré, Meloca, Roma, Ismael, & Silva, 2018)
19.	A novel trust model based on SLA and behavior evaluation for clouds	(Z. Tan et al., 2016)
20.	Elasticity and scalability centric quality model for the cloud	(Abdeladim, Baina, & Baina, 2014)

Los artículos clasificados por Carrot2 pueden estar en más de un subnodo, esta relación es establecida por el algoritmo de Lingo. En la Tabla 24 se presentan los artículos del nodo "Evaluation Model (20)" y el subnodo al cual fue relacionado por Lingo.

Tabla 24. Relación entre subnodos. Artículos nodo "Evaluation Model (20)

Nro.	Nombre de artículo	Referencia	Subnodos															
			Evaluate what Quality Attribute (6)	Cloud Providers (5)	QoS Evaluation (5)	Metric Model (4)	Security and Reliability Function (4)	Trusted Cloud Service (4)	Ability (2)	Availability (2)	Efficiency (2)	Elasticity (2)	Security and Privacy (2)	Service Level (2)	Software Quality (2)	Other Topics (1)		
1	An Approach to IdM with Privacy in the Cloud.	(Werner et al., 2015)		x												x		
2	Quality Model for Evaluating Platform as a Service in Cloud Computing.	(Olokunde et al., 2012)															x	
3	A Quality Model for Evaluating Software-as-a-Service in Cloud Computing.	(Lee et al., 2009)	x			x	x											
4	Towards Service Evaluation and Ranking Model for Cloud Infrastructure Selection.	(Khan & Qamar, 2016)		x												x		
5	Quality evaluation and best service choice for cloud computing based on user preference and weights of attributes using the analytic network Process.	(Choi & Jeong, 2014)					x					x						
6	Cloud service evaluation method-based Multi-Criteria Decision-Making: A systematic literature review.	(Alabool et al., 2018)	x		x	x			x									
7	Designing an efficient methodology based on Entropy-TOPSIS for evaluating efficiency of cloud services	(R. R. Kumar & Kumar, 2017)										x						
8	The evaluation system for cloud service quality based on SERVQUAL	(Hu & Zhang, 2013)															x	
9	QoS evaluation for web services in cloud computing	(Nadanam & Rajmohan, 2012)											x					
10	Software metric evaluation on cloud based applications	(Phani Sheetal & Ravindranath, 2018)															x	
11	Research on the measurement and evaluation of trusted cloud service	(Ma et al., 2018)	x															
12	Modeling Correlation between QoS Attributes for Trust Computation in Cloud Computing Environments	(Mrabet et al., 2017)																
13	A QoS Evaluation Model for Test-Bed in the Cloud Computing Environment	(Song et al., 2012)																
14	An Evaluation Model for Selecting Cloud Services from Commercially Available Cloud Providers	(Wagle et al., 2016)		x														
15	A Stochastic Model to Investigate Data Center Performance and QoS in IaaS Cloud Computing Systems	(Bruneo, 2014)																
16	Semantically Equivalent Model for Quality Evaluation	(Kara et al., 2017)	x															
17	Optimal Negotiation of Service Level Agreements for Cloud-Based Services through Autonomous Agents	(Yaqub et al., 2014)		x	x												x	
18	An empirical study for evaluating the performance of multi-cloud APIs	(Ré et al., 2018)	x															
19	A novel trust model based on SLA and behavior evaluation for clouds	(Tan et al., 2016)		x													x	
20	Elasticity and scalability centric quality model for the cloud	(Abdeladim et al., 2014)	x		x												x	

El detalle de cada uno de los nodos internos del clúster de nivel 1 y la relación de artículos se encuentra en el Apéndice A.

4.3.8. Extracción de datos

La fase de extracción de datos tiene tres entradas: 1) Las características propuestas por el SMI, 2) los clústeres generados por Carrot2 y 3) los 119 trabajos que aprobaron la fase de evaluación de calidad.

Para la extracción de datos se utilizó en primera instancia un instrumento en Excel, sin embargo, esta estrategia no dio resultados por la complejidad de las relaciones entre los datos obtenidos. De acuerdo con Reis, Costa, & de Souza (2016) existen herramientas de análisis cualitativo que “pueden servir para análisis de transcripción, escritura y anotación, codificación e interpretación de texto, abstracción recursiva, búsqueda y análisis de contenido, análisis del discurso, mapeo de datos, metodología de teoría fundamentada, entre otros tipos de análisis”. De acuerdo con Ferreira, Moreno, Brandao, & Cerqueira (2016), entre las mejores herramientas para gestionar investigaciones está Atlas.ti. Aunque Atlas.ti es una herramienta ideal para extracción de datos, no permite administrar una estructura de información jerárquica que sirva para organizar los datos para construir la taxonomía. Descartadas las dos primeras alternativas, fue necesario construir una herramienta, para apoyar la construcción de la taxonomía desde el proceso de extracción de datos. A continuación, se describe la estructura de la herramienta.

4.3.8.1. Modelo recursivo

Una taxonomía jerárquica permite clasificar la información en taxones o grupos. Cada taxón puede tener una clasificación interna y así sucesivamente (Hlava, 2014). Por tal razón, se requiere una estructura que permita apoyar este tipo de modelado, de tal manera que la extracción de datos sea eficiente. La Figura 14 presenta el modelo de objetos propuesto para el instrumento de extracción.

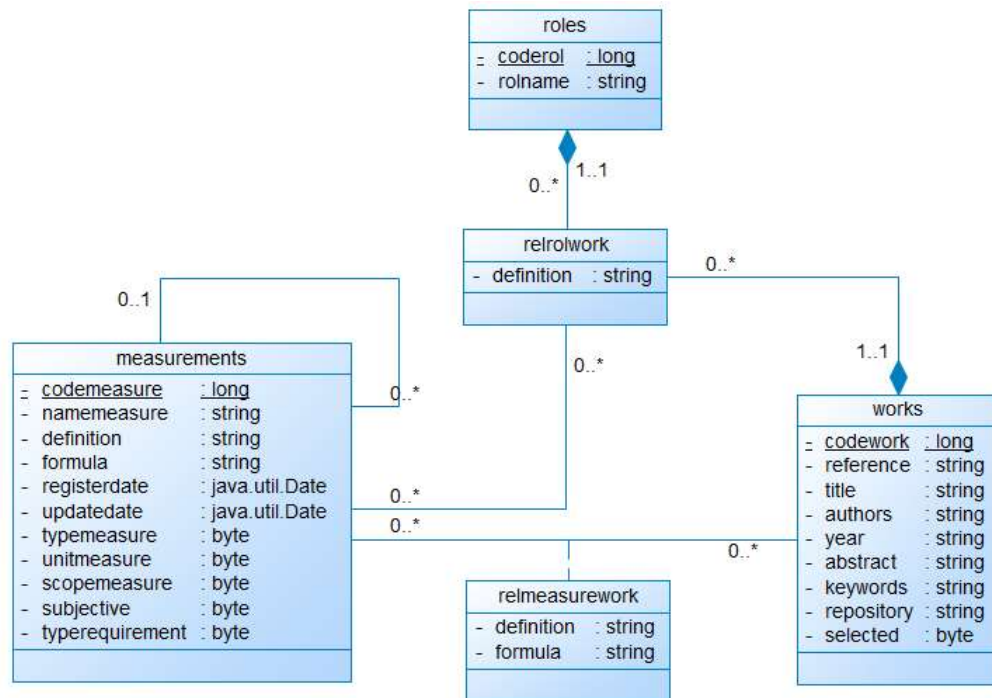


Figura 14. Diagrama de clases para extracción de datos

La clase “*measurements*”, entendida como las características, permite almacenar toda la información que puede llegar a tener un taxón. El enfoque recursivo facilita que una característica tenga subcaracterísticas con la misma estructura y así sucesivamente. Esto permite almacenar una estructura en forma de árbol útil para la construcción de la taxonomía. La clase “*works*” se asocia a los trabajos seleccionados después de la fase de calidad. La clase “*roles*” hace referencia a los posibles actores en la computación en la nube identificados en los trabajos seleccionados. La clase “*relmeasurework*” contiene la definición y la fórmula de una característica en un trabajo en particular. La clase “*relrolwork*” relaciona un rol, con un trabajo y una medida. A partir de este modelo de objetos se generó el modelo físico, el cual permitió el registro y almacenamiento de la información extraída en una base de datos relacional.

4.3.8.2. Diseño de la herramienta

Tomando como base el modelo de objetos recursivo, se desarrolló una interfaz web capaz de permitir la extracción de los datos y la organización de la taxonomía de forma dinámica, esta herramienta permite: almacenar, visualizar y buscar atributos de calidad de la taxonomía, convirtiéndose en un instrumento transversal para todo el proceso. La Figura 15 presenta la interfaz web utilizada para la organización de la estructura jerárquica.

Order	characteristic	Subchar
1	📁 Taxonomy	9
1.1	▶ 📁 node 1	15
1.2	📁 node 2	6
1.2.1	📄 node 2.1	
1.2.2	📄 node 2.2	
1.3	📁 node 3	7
1.3.1	📁 node 3.1	10
1.3.1.1	📄 node 3.1.1	
1.3.1.2	📄 node 3.1.2	
1.3.1.3	📄 node 3.1.3	
1.3.2	📄 node 3.2	
1.4	▶ 📁 node 4	5

Edit	[F2]
Delete	[Del]
New sibling	[Ctrl+N]
New child	[Ctrl+Shift+N]
Cut	Ctrl+X
Copy	Ctrl+C
Paste as child	Ctrl+V

Figura 15. Interfaz Web del instrumento - árbol

Cada taxón o característica tiene opciones para actualizar, eliminar, crear nuevas características al mismo nivel o a un nivel interno. La principal ventaja de la herramienta es que a medida que se realiza la extracción de los datos, la taxonomía va quedando organizada. Adicionalmente, a la estructura de árbol, el instrumento cuenta con opciones para edición y búsqueda de información. La Figura 16 presenta los elementos adicionales del instrumento.

Cancel

Details
Related works
Associated role

Definition
 Taxonomy: no definition

Type

Unit

Scope

Category

Requirement

Formula
 no formula

Figura 16. Interfaz Web del instrumento – opciones adicionales

4.3.8.3. Resultados de la extracción

Los resultados de la utilización del instrumento construido para la extracción de datos se encuentran en el capítulo 5 PRESENTACIÓN DE LA TAXONOMÍA.

4.4. Resumen de la ejecución del protocolo

(Kitchenham, 2007) planteó un proceso sistemático para la revisión de literatura, este proceso presenta los lineamientos necesarios para realizar un estudio con características de calidad que permitan realizar seguimiento y control de una investigación. El proceso propuesto por (Kitchenham, 2007) fue la base para la construcción del protocolo que permitió la construcción de la taxonomía planteada en esta investigación.

El protocolo inicia con la definición de la cadena de búsqueda y continua con la selección de los repositorios digitales. Para evitar doble procesamiento de la información, se realizó un filtro para eliminar estudios que podrían estar repetidos en los repositorios. El filtro para eliminar estudios repetidos se limitó al título, por lo tanto, un carácter o un cambio menor en el título daba como resultado la no exclusión del estudio.

Para realizar la preselección de estudios se llevaron a cabo dos etapas: 1) el procesamiento automático de estudios realizado con la herramienta VOSviewer y 2) el procesamiento manual de los estudios. El contraste de los dos procesos permitió realizar una preselección más precisa de los trabajos.

Posteriormente se realizó la evaluación de calidad de los estudios, se definieron dos factores de obligatorio cumplimiento para que un trabajo aprobara esta fase. Finalizada la fase de selección definitiva de trabajos a procesar, se continuó con la extracción de los datos, en esta fase se incluyó la línea de base sugerida por ANSI/NISO (2005): SMI y clústeres procesados con Carrot2.

Durante el proceso de extracción se realizó el análisis de las relaciones entre las características y atributos encontrados, de esta forma se logró construir la taxonomía de manera gradual.

La Figura 17 presenta el paso a paso de cada una de las fases desarrolladas en el protocolo.

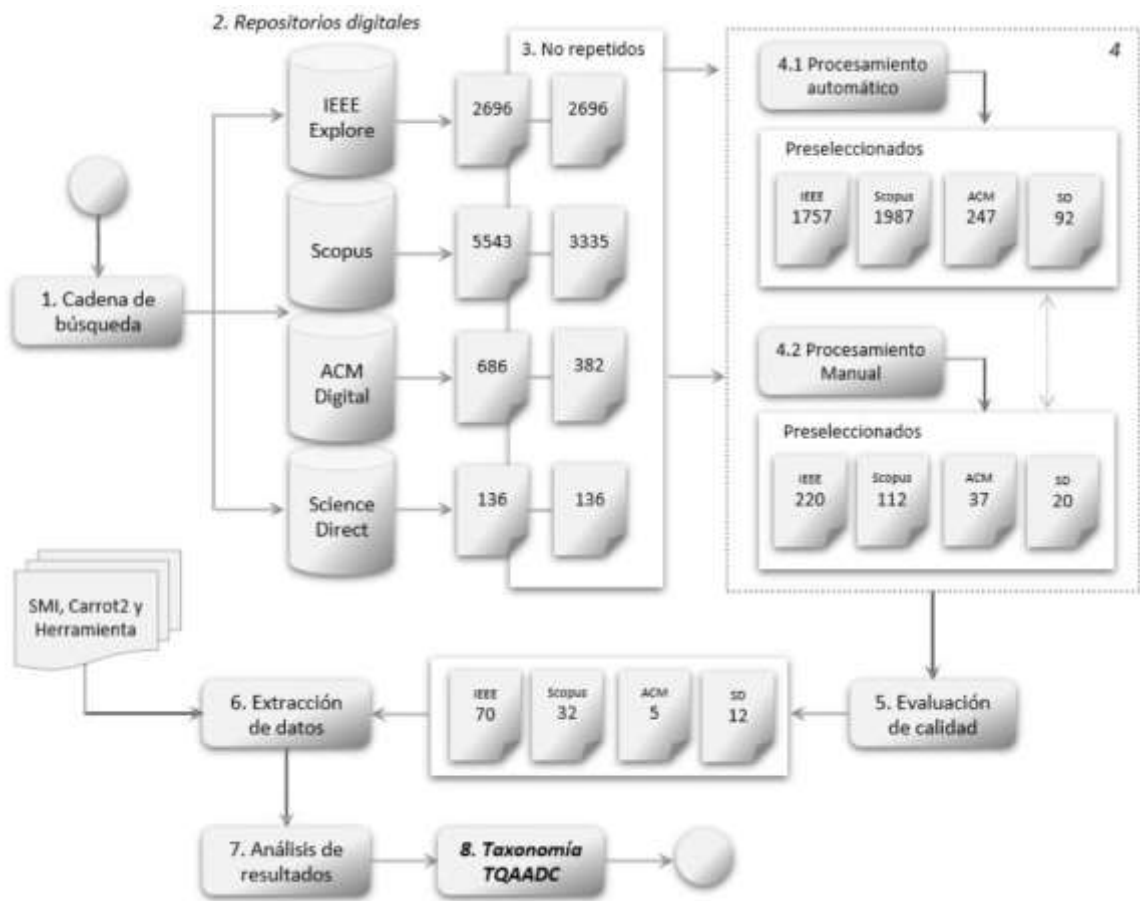


Figura 17. Fases del protocolo desarrolladas

El esfuerzo principal del trabajo realizado está orientado a la organización de conocimiento, obteniendo como producto principal la taxonomía sobre atributos de calidad para aplicaciones a ser desplegadas en la nube. A continuación, se presentan algunos hallazgos secundarios relacionados con los 119 estudios que fueron finalmente utilizados para la construcción de la taxonomía. La Figura 18 presenta la cantidad de estudios seleccionados por país de origen.

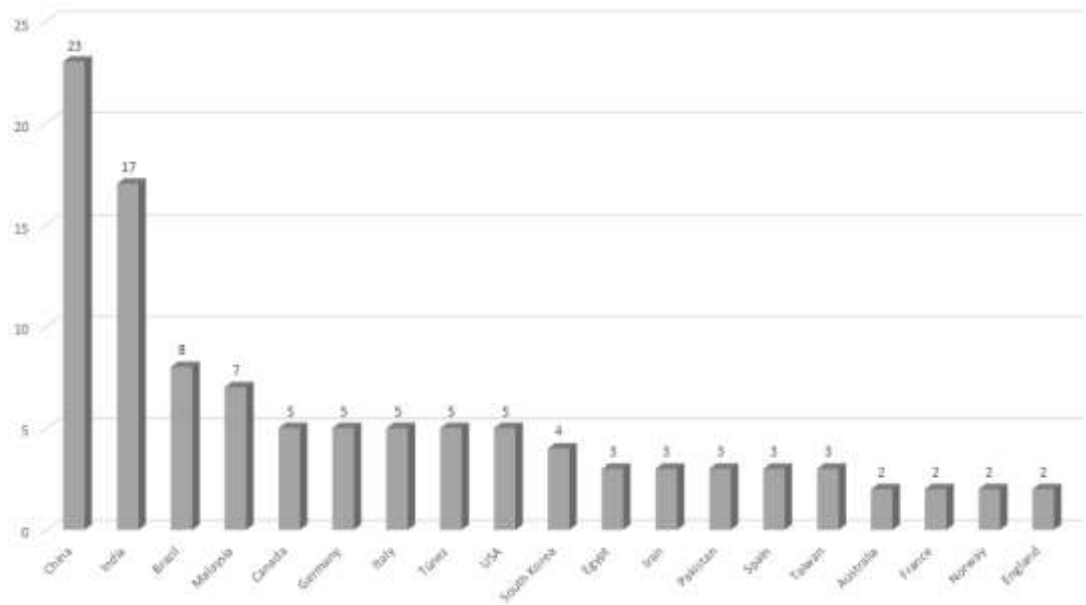


Figura 18. Cantidad de artículos por país de origen del autor

China, India, Brasil y Malasia aportaron el 46% de los estudios utilizados en la presente investigación, siendo los países con más aportes. Un segundo grupo compuesto por Canadá, Alemania, Italia, Túnez y Estados Unidos aportaron el 21% de los estudios. El 33% restante fueron estudios de diferentes países con aportes entre 1 o 2 estudios.

En la Figura 19 se presenta la ventana de observación de los estudios finalmente seleccionados.

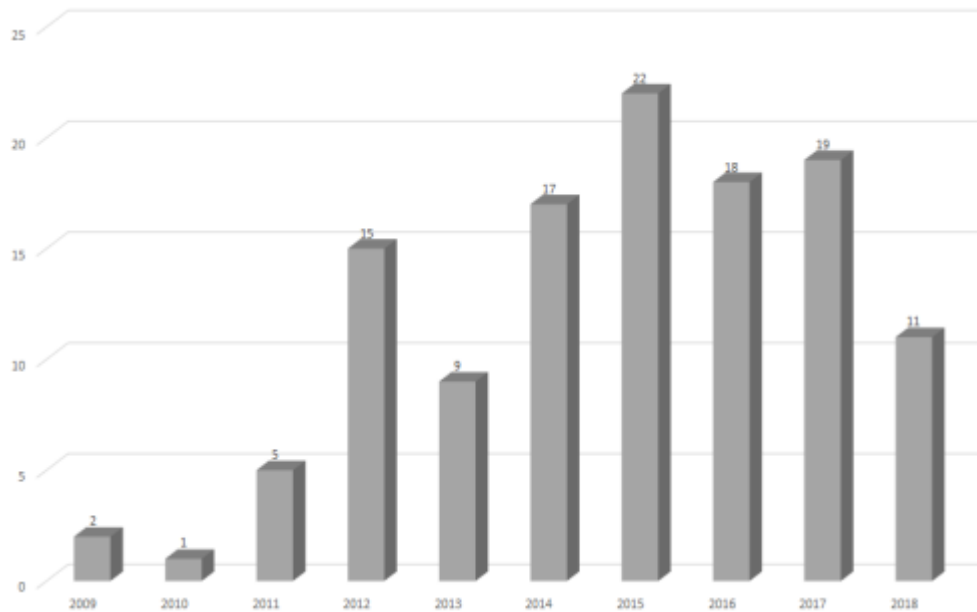


Figura 19. Ventana de observación de los estudios seleccionados

La investigación tomó como referencia la ventana de observación entre 2006 y 2018, finalmente el mayor volumen de los estudios seleccionados está entre 2012 y 2018.

4.5. Amenazas a la validez del trabajo realizado

Uno de los principales problemas en la revisión sistemática de literatura es el sesgo de información (Drucker, Fleming, & Chan, 2016), un proceso puede incluir o no trabajos que alteren significativamente los resultados del estudio. Esa inclusión selectiva se mitigó en esta investigación en dos formas, por un lado, se generó una cadena de búsqueda que permitiera una gran cobertura en la obtención de trabajos relacionados. Por otro lado, los repositorios de alto impacto seleccionados generan de manera automática los registros con los estudios a analizar, éstos hacen parte de la base documental de la investigación.

Otro tipo de amenaza es el sesgo de informe (Drucker et al., 2016), este surge cuando no se tiene claridad sobre cómo, cuándo y por qué se realizó la investigación. Para mitigar esta amenaza, se presenta un apartado completo que describe en detalle cómo se realizó, en que momentos y cuáles fueron las preguntas de investigación de la revisión.

Adicionalmente, una revisión de literatura debe realizarse sobre fuentes que tengan validez (Barité, 2009), para mitigar este problema, esta investigación acoge el concepto de garantía literaria, puesto que cada uno de los repositorios digitales seleccionados tiene procesos de verificación y validación por pares externos, quienes son los encargados de avalar una publicación de alto impacto.

Otro problema en la revisión de literatura es la selección, evaluación de la calidad y extracción de datos. En esta investigación se mitigó esta amenaza de la siguiente forma: El investigador principal realizaba la selección, evaluación y extracción de datos, posteriormente, dos revisores adicionales realizaban las mismas actividades para evidenciar similitudes e inconsistencias.

5. PRESENTACIÓN DE LA TAXONOMÍA

5.1. Principios

El objetivo fundamental de la presente taxonomía es apoyar la organización del conocimiento alrededor de la temática atributos de calidad para aplicaciones a ser desplegadas en la nube. El presente trabajo cumple con los principios básicos propuestos por ANSI/NISO (2005):

1. Suministrar un vocabulario que se pueda indexar y recuperar: La taxonomía se encuentra publicada en un servidor de acceso público, su estructura tiene una codificación única para cada nodo y suministra un sistema de recuperación de contenido basado en un patrón de búsqueda suministrado por el usuario. La URL en donde se encuentra disponible la herramienta que permite la visualización de la taxonomía es: <https://www.tgaca.org/>
2. Promover la uniformidad en los formatos utilizados para cada término y en la asignación de los términos: La utilización de la herramienta en línea permitió automatizar el proceso de registro, presentación y uniformidad en la asignación de cada término.
3. Indicar las relaciones entre los términos: La estructura seleccionada para la presentación de la taxonomía es jerárquica, de este tipo de presentación se utilizaron dos tipos de relaciones: genérica y todo-parte.
4. Servir como ayuda para localizar objetos de contenido: El repositorio digital es público y permite acceder a la totalidad de las características de calidad identificadas. La funcionalidad para descargar los documentos de las fuentes primarias de la investigación no se encuentra habilitada.

5.2. Tipo de relaciones utilizadas

Existen tres tipos de relaciones para definir taxonomías (ANSI/NISO, 2005): 1) equivalencia, 2) jerarquía y 3) asociación. En esta investigación se utilizó el tipo de relación jerárquica, basada en niveles superordinados y subordinados. Los superordinados o padres (término utilizado en inglés: BT - broader term) representan una característica o un nodo completo. Los subordinados o hijos (Término utilizado en inglés: NT - narrower term) representan subcaracterísticas en la presente taxonomía. De los tres tipos de relaciones definidos para una relación jerárquica, en esta investigación se utilizaron dos:

5.2.1. Relación genérica

Las relaciones genéricas establecen un enlace directo entre la clase principal y sus miembros. De acuerdo con ANSI/NISO (2005), toda taxonomía jerárquica que utilice este tipo de relación en alguno de sus BT/NT, debe pasar la prueba "ALL and SOME" y se

puede representar con las siglas BTG/NTG, donde la “G” representa genérico. Un ejemplo de este tipo de relación encontrado en la taxonomía se presenta en la Figura 20.



Figura 20. Prueba All and Some

En Figura 20 se presentan dos ejemplos, para el caso de la relación entre Reliability y Stability, la mano con pulgar arriba en el círculo verde representa la existencia de una relación entre Reliability y Recoverability, la mano con pulgar abajo en el círculo naranja, representa que no existe relación entre estos atributos. Cada relación de la taxonomía fue verificada teniendo en cuenta el criterio de prueba “ALL and SOME”.

5.2.2. Relación todo - parte

Este tipo de relación hace referencia a un término que incluye a otro sin importar el contexto, esto quiere decir que los términos pueden ser organizados de acuerdo con una jerarquía lógica. La representación en códigos que plantea ANSI/NISO (2005) es BTP (Broader Term Partitive) y NTP (Narrower Term Partitive). Un ejemplo de este tipo de relación encontrado en la taxonomía se presenta en la Figura 21.



Figura 21. Relación Todo - Parte

5.3. Tipo de presentación

De acuerdo con ANSI/NISO (2005) existe una gran variedad de estructuras para presentar vocabularios controlados y taxonomías: 1) listas alfabéticas, 2) presentación permutada, 3) presentación detallada, 4) presentación jerárquica y 5) presentación gráfica de árbol. La taxonomía desarrollada acoge las recomendaciones de la ANSI/NISO (2005) tomando como referencia la presentación jerárquica y su extensión en forma de estructura de árbol. En la estructura de árbol cada característica es asignada a un grupo, organizado alfabéticamente y con una representación visual jerárquica.

5.4. Una nueva taxonomía sobre atributos de calidad en la nube

La taxonomía construida tiene por nombre “Taxonomía de Atributos de Calidad en Aplicaciones para la Nube”, TQACA por su nombre en inglés “Taxonomy on Quality Attributes for Cloud Applications”. Es el resultado de la ejecución del protocolo diseñado en la presente investigación.

El instrumento utilizado en la fase de extracción de datos permitió construir la taxonomía de manera paralela desde esa fase. De acuerdo con las recomendaciones propuestas por el Centro de Interoperabilidad Semántica de Europa (Fraunhofer ISST, 2009), se utilizó como línea de base el SMI y los clústeres generados por Carrot2. La Figura 22 presenta el resultado del primer nivel de la taxonomía propuesta en la presente investigación.

Order	Categories / Attributes	Nodes
1	📁 Taxonomy	9
1.1	▷ 📁 Accountability	14
1.2	▷ 📁 Agility	8
1.3	▷ 📁 Assurance	27
1.4	▷ 📁 Empathy	5
1.5	▷ 📁 Financial	6
1.6	▷ 📁 Performance	17
1.7	▷ 📁 Security and privacy	24
1.8	▷ 📁 Usability	20
1.9	▷ 📁 User Experience	1

Figura 22. Categorías de primer nivel de TQACA

TQACA incluye dos nuevas categorías no definidas en SMI, resaltadas en color verde en la Figura 22, además, contiene un 42 % de nuevas y diferentes características de calidad. La estructura multinivel de TQACA permite organizar las características de calidad en cuatro niveles (abuelo, padre, hijo, nieto), a diferencia de SMI que lo hace en dos (padre, hijo). La coherencia y la cohesión entre las características se realizó de dos formas: 1) Análisis de los estudios seleccionados, realizado por el investigador principal y su respectiva verificación por parte de 2 investigadores externos que acompañaron el proceso extracción de información y 2) Con base en el análisis realizado a los clústeres generados por la herramienta Carrot2.

A continuación, se presentan el resumen de las definiciones para cada una de las características encontradas, los trabajos relacionados y sus fuentes. En el Apéndice B, se encuentra el detalle de cada característica y atributo.

5.4.1. Característica “1.1 Accountability”

1.1	Accountability	11
1.1.1	Compliance	
1.1.2	Contracting experience	
1.1.3	Ease of doing business	
1.1.4	Governance	
1.1.5	Provider certifications	
1.1.6	Provider contract /SLA verification	
1.1.7	Provider personnel requirements	
1.1.8	Provider supply chain	
1.1.9	Provider support	3
1.1.9.1	Customer support facility	
1.1.9.2	Operating Systems Support	
1.1.9.3	Platforms Supported	
1.1.10	Sustainability	
1.1.11	Technology Change	

Esta categoría ha sido abordada en seis trabajos: (M. Z. Khan & Qamar, 2015), (Shin & Huh, 2015), (Siegel & Perdue, 2012), (Yadav & Goraya, 2018), (Nuñez, Fernández-Gago, & Luna, 2016), (Tchernykh, Schwiegelsohn, Talbi, & Babenko, 2016)

Contiene los atributos que permiten medir propiedades relacionadas con la forma como el proveedor organiza y ofrece los servicios que tiene en la nube.

Las propiedades no necesariamente están relacionadas con el servicio que presta el proveedor. Los atributos relacionados con la característica *Accountability* son descritos en la Tabla 25.

Tabla 25. Atributos de “Accountability”

Atributo	Definición	Trabajos relacionados
1.1.1 Compliance	Cumplimiento de los estándares, procesos y políticas a los cuales el proveedor se compromete al ofrecer los servicios.	(G. Raj, Sarfaraz, & Singh, 2014), (M. K. Naseer et al., 2014), (M. Ahmed, Liu, Hardy, & Yuan, 2014), (Salleh, Bahari, & Zakaria, 2017), (Nuñez et al., 2016)
1.1.2 Contracting experience	Indicadores del esfuerzo del cliente y la satisfacción con el proceso de suscripción de los acuerdos necesarios para utilizar un servicio.	(Baranwal & Vidyarthi, 2014), (Monir, AbdelAziz, AbdelHamid, & El-Horbaty, 2015), (Assis & Bittencourt, 2016)
1.1.3 Ease of doing business	Satisfacción del cliente al realizar negocios con un proveedor de servicios.	(Assis & Bittencourt, 2016)
1.1.4 Governance	Proceso usado por el proveedor de servicios de nube para gestionar las expectativas, problemas y desempeño del servicio con el cliente.	(Ghosh & Ghosh, 2012)
1.1.5 Provider certifications	El proveedor de servicios en la nube tiene certificaciones activas de estándares relevantes para los requerimientos de sus clientes. Identifica el cumplimiento de los estándares por parte del Cloud Service Provider (CSP). Evalúa los antecedentes, calificaciones, y legitimidad del CSP en el manejo de seguridad de la información, privacidad, atención médica, educación, etc. Consideraciones incluidas: 1) El CSP y el servicio están certificados? 2) ¿Los consumidores pueden verificar la validez de las certificaciones?	(Baranwal & Vidyarthi, 2014), (Abuhussein, Alsubaei, Shiva, & Sheldon, 2016), (Baranwal & Vidyarthi, 2016)
1.1.6 Provider contract / SLA verification	El CSP tiene disponible a los clientes los SLAs para gestionar el servicio y mitigar riesgos de fallas. Identifica que el servicio es ejecutado conforme a los SLA. Consideraciones: 1) ¿El CSP considera la complejidad de la arquitectura de la nube en los SLA? 2) Los SLAs consideran los diferentes impactos ante la interrupción del servicio por razones comerciales?	(Abuhussein et al., 2016), (G. Raj et al., 2014), (Monir et al., 2015), (Alkhanak, Lee, & Khan, 2015), (Nuñez et al., 2016), (Alkhanak et al., 2015)

Atributo	Definición	Trabajos relacionados
1.1.7 Provider personnel requirements	<p>Grado de competencias, experiencia, educación, y certificaciones del personal del CSP, necesarias para prestar un servicio de manera efectiva.</p> <p>Evalúa las políticas, estrategias y procedimientos del CSP para prevenir, detectar y/o responder a ataques maliciosos realizados sobre el servicio, por una persona con acceso autorizado. Consideraciones incluidas: 1) ¿El CSP tiene un programa formalizado de amenazas internas? 2) ¿El CSP realiza una verificación de antecedentes de los empleados?</p>	(Abuhussein et al., 2016)
1.1.8 Provider supply chain	<p>El CSP asegura que todos los SLA que deben ser compatibles con los proveedores, lo son realmente.</p> <p>Evalúa las políticas, estrategias y procedimientos que el CSP utiliza para validar la legitimidad y compatibilidad de la seguridad y privacidad del hardware y software comprado, para los servicios en la nube.</p> <p>Consideraciones incluidas: 1) ¿El CSP compra/arrenda hardware y software de una fuente legítima? 2) ¿El CSP compra y arrenda hardware y software probados, certificados y que cumplen con los estándares?</p>	(Abuhussein et al., 2016)
1.1.9 Provider support	<p>Medida en que el CSP dispone servicios para dar soporte al cliente, incluidas respuesta a preguntas sobre el servicio y la solución o corrección de problemas que puedan surgir.</p> <p>Se refiere a la disponibilidad del CSP para compensar a los consumidores de los servicios de la nube por posibles fallas específicas en el servicio.</p> <p>Consideraciones incluidas: 1) ¿E CSP brinda seguro para los servicios? 2) ¿CSP tiene más de un plan de seguro?</p>	(Abuhussein et al., 2016), (Hu & Zhang, 2013)
1.1.9.1 Customer Support facility	<p>Tipo, tiempo de respuesta y cobro que se realiza al cliente por un soporte, son factores importantes que definen el soporte al cliente.</p> <p>Los usuarios nuevos prefieren un proveedor con un buen sistema de soporte.</p> <p>Algunos proveedores ofrecen un servicio de atención al cliente</p>	(Baranwal & Vidyarthi, 2014), (G. Raj et al., 2014), (M. K. Naseer et al., 2014), (Monir et al., 2015), (Baranwal & Vidyarthi, 2016)

Atributo	Definición	Trabajos relacionados
	<p>gratuito, pero en su mayoría los CSP cobran el soporte. GoGrid proporciona soporte telefónico gratuito 24/7 y soporte premium gratuito 24/7. Amazon AWS proporciona soporte premium con los siguientes tiempos de respuesta de acuerdo con el nivel de prioridad (urgente: 1 hora, alta: 4 horas hábiles, normal: 1 día hábil, baja: 2 días hábiles). La métrica puede incluir elementos como soporte telefónico gratuito 24/7, soporte urgente, soporte básico, soporte bajo, herramientas de diagnóstico, etc.</p>	
<p>1.1.9.2 Operating system support</p>	<p>Los proveedores soportan diferentes sistemas operativos como MacOS X, Windows y Open SUSE Linux, etc. Es posible que entre un proveedor y otro se admitan diferentes sistemas operativos. Por ejemplo, Windows Azure es compatible con el sistema operativo Windows y Linux, mientras que GoGrid es compatible con el servidor Windows 2003/2008, Red Hat Linux 5.1/5.4, etc. Diferentes aplicaciones requieren diferentes sistemas operativos.</p>	<p>(Baranwal & Vidyarthi, 2014), (Baranwal & Vidyarthi, 2016)</p>
<p>1.1.9.3 Platforms supported</p>	<p>Así como los sistemas operativos, diferentes proveedores soportan diferentes tipos de plataformas. Por ejemplo, CloudSigma es compatible con Java, PHP, WinDev, Dot Net. Mientras que Firehost admite Ruby, Java, PHP y Dot Net. Diferentes aplicaciones requieren soporte de plataforma diferente.</p>	<p>(Baranwal & Vidyarthi, 2014), (De Benedetti, Durso, Messina, Pappalardo, & Santoro, 2015), (Baranwal & Vidyarthi, 2016)</p>
<p>1.1.10 Sustainability</p>	<p>La sostenibilidad tiene tres dimensiones; Impacto económico, social y ambiental del CSP. El impacto ambiental se puede medir utilizando métricas como la emisión de carbono, el uso del agua y el consumo de recursos. El impacto económico se puede medir utilizando métricas como el costo de energía de los servidores, el almacenamiento, el costo de las instalaciones de red y el costo de soporte. El impacto social se puede medir utilizando métricas como el desarrollo económico y la estabilidad sociopolítica.</p>	<p>(Baranwal & Vidyarthi, 2014), (Phani Sheetal & Ravindranath, 2018), (Baranwal & Vidyarthi, 2016)</p>
<p>1.1.11 Technology change</p>	<p>Mide la preparación del CSP para hacer frente a los problemas de seguridad y privacidad que pueden ocurrir como resultado de</p>	<p>(Abuhussein et al., 2016)</p>

Atributo	Definición	Trabajos relacionados
	la evolución de la tecnología o la depreciación de esta. Consideraciones incluidas: 1) ¿Puede el CSP cambiar las características de seguridad en caso de que surja una nueva tecnología o pase a ser obsoleta, sin afectar el servicio? 2) ¿El CSP guía a los consumidores a tomar decisiones informadas si la tecnología, los atributos o riesgos de seguridad y privacidad?	

5.4.2. Característica “1.2 Agility”

1.2	Agility	5
1.2.1	Adaptability	
1.2.2	Flexibility	
1.2.3	Portability	
1.2.4	Responsiveness	3
1.2.4.1	Elasticity	4
1.2.4.1.1	Mean Time To Quality Repair	
1.2.4.1.2	Number of SLO violations	
1.2.4.1.3	VM Shutdown Time	
1.2.4.1.4	VM Startup Time	
1.2.4.2	Interaction	
1.2.4.3	Timeliness	
1.2.5	Scalability	5
1.2.5.1	Customizability	
1.2.5.2	Scalability Range	
1.2.5.3	Scalability Speed	
1.2.5.4	Virtual infrastructure automation	
1.2.5.5	Virtualization Technique	

Esta categoría ha sido abordada en seis trabajos: (M. Z. Khan & Qamar, 2015), (Shin & Huh, 2015), (Monir et al., 2015), (Siegel & Perdue, 2012), (Baliyan & Kumar, 2013), (Yadav & Goraya, 2018)

Indica el impacto de un servicio sobre la capacidad de un cliente para cambiar de dirección, estrategia o táctica de forma rápida y con una interrupción mínima.

Los atributos relacionados con la característica Agility son descritos en la Tabla 26.

Tabla 26. Atributos de “Agility”

Atributo	Definición	Trabajos relacionados
1.2.1 Adaptability	Capacidad del CSP para responder a solicitudes inesperadas de las organizaciones, ajustando o modificando los cambios en sus servicios. La adaptabilidad se ve cada vez más como un factor importante para mejorar la eficiencia de una organización y lograr el éxito económico.	(Grandhi & Wibowo, 2016), (Monir et al., 2015), (Nadanam & Rajmohan, 2012), (Muñoz-Escoí & Bernabéu-Aubán, 2017)
1.2.2 Flexibility	Capacidad de eliminar o adicionar funciones predefinidas de un servicio.	(Techio & Misaghi, 2015), (Oh et al., 2011), (Song et al., 2012), (Nadanam & Rajmohan, 2012), (Sodhi & Prabhakar, 2012), (S. Singh & Chana, 2015)
1.2.3 Portability	Desde la perspectiva del cliente, es la habilidad de un cliente para trasladar fácilmente un servicio de un CSP a otro con una interrupción mínima. Desde la perspectiva del proveedor, es la capacidad del CSP para mover los datos de un sistema en la nube a otro, sin degradar el rendimiento. Se dice que un CSP ofrece portabilidad durante un período, si sigue siendo confiable y efectivo en el caso de saltar de un sistema de nube a otro.	(Mrabet et al., 2017b), (G. Raj et al., 2014), (Medeiros, Ivaki, Costa, & Vieira, 2017), (Techio & Misaghi, 2015), (S. Singh & Chana, 2015), (White, Nallur, & Clarke, 2017), (Olokunde et al., 2017), (Shin & Huh, 2016), (Choi & Jeong, 2014)
1.2.4 Responsiveness	Los clientes exigen respuesta oportuna en el servicio. Por tal razón la capacidad de respuesta del servicio en la nube se señala como un atributo importante para garantizar la calidad. Puede definirse como el grado en que el proveedor de servicios en la nube está listo y puede ayudar a los clientes al momento de proporcionar servicios oportunos y rápidos. La capacidad de respuesta (Responsiveness RESP), es la rapidez de los servicios ante una solicitud durante un intervalo de tiempo. En lugar de un tiempo máximo posible desde la perspectiva del sistema, utilizamos el tiempo máximo aceptable desde el punto de vista del usuario final para formular la capacidad de respuesta. Para los servicios en la nube, una medición desde el punto de	(M. Naseer & Nazar, 2017), (Zhou et al., 2015), (X. Zheng et al., 2014), (Bruneo, 2014), (Z. Wang et al., 2015), (Sodhi & Prabhakar, 2012), (Mohamed, 2016)

Atributo	Definición	Trabajos relacionados
	<p>vista de un usuario final es mejor que la de una perspectiva del sistema.</p> <p>En la actualidad, pocos servicios de nube mencionan la capacidad de respuesta en sus SLA. Como los servicios en la nube recomiendan el cálculo, en un sentido general, como un servicio, donde una respuesta rápida indica un alto rendimiento, los consumidores valoran un servicio preciso y rápido.</p>	
1.2.4.1 Elasticity	<p>Capacidad de un CSP para ajustar el consumo de recursos de un servicio, a una velocidad suficientemente rápida para satisfacer la demanda del cliente.</p> <p>La elasticidad se puede entender por sus dos elementos fundamentales: tiempo y costo. Tiempo significa cuánto tiempo tarda un CSP en proveer o retirar recursos y el costo significa si el CSP cobra por hora o por minuto.</p>	<p>(Baranwal & Vidyarthi, 2014), (Zhou et al., 2015), (Abdeladim et al., 2014), (Ghahramani, Zhou, & Hon, 2017), (X. Zheng et al., 2014), (Sebastian Lehrig, Eikerling, & Becker, 2015), (Dustdar, Guo, Satzger, & Truong, 2011), (Klems et al., 2012), (Muñoz-Escóí & Bernabéu-Aubán, 2017), (Baranwal & Vidyarthi, 2016)</p>
1.2.4.1.1 Mean time to quality repair	<p>MTTQR. Medida de rapidez con la que un sistema puede adaptarse a los cambios en la carga de trabajo. Es una medida de la elasticidad y depende de un delta de la carga de trabajo, es decir, el aumento/disminución entre dos cargas de trabajo.</p> <p>MTTQR define el tiempo medio que necesita un sistema para restablecer su Service Level Objective (SLO), cuando la carga de trabajo aumenta/disminuye para un delta de carga de trabajo definido, especificado como factor (número real). Por lo tanto, MTTQR se mide en unidades de tiempo. Dado que define un tiempo medio, MTTQR es específico para un marco de tiempo definido en el que se calcula el promedio.</p>	<p>(Becker, Lehrig, & Becker, 2015), (S. Lehrig & Becker, 2015)</p>
1.2.4.1.2 Number of SLO violations	<p>NSLOV. El número de violaciones de SLO en un intervalo de tiempo definido. Esta métrica mide la elasticidad de un sistema.</p> <p>La carga de trabajo delta también se especifica como un factor (número real). NSLOV refleja con qué frecuencia un sistema viola sus SLO, cuando la carga de trabajo cambia a una tasa determinada, medida como un número real.</p>	<p>(Becker et al., 2015), (S. Lehrig & Becker, 2015)</p>

Atributo	Definición	Trabajos relacionados
1.2.4.1.3 VM Shutdown time	Tiempo consumido para apagar y retirar máquinas virtuales.	(Qu & Buyya, 2014)
1.2.4.1.4 VM Startup time	Tiempo consumido para asignar, arrancar y configurar máquinas virtuales.	(Qu & Buyya, 2014)
1.2.4.2 Interaction	Grado en que un servicio en la nube responde a los clientes de manera apropiada. Esto corresponde a la rapidez con la cual un servicio responde ante una solicitud de un cliente en un intervalo de tiempo.	(Zhou et al., 2015)
1.2.4.3 Timeless	Grado en que la velocidad de respuesta de un servicio en la nube satisface las necesidades del cliente. Habilidad de un sistema para suministrar información a tiempo cuando los usuarios necesitan acceder a él. Mide el tiempo total necesario para completar una tarea. Esto se mide como retraso, latencia o tiempo para completar. Esta métrica también mide la hora de inicio y la fecha límite para una tarea.	(Zhou et al., 2015), (Ghahramani et al., 2017), (Kim, Mohan, & Ramesh, 2014), (Mrabet, Saied, & Saidane, 2017a), (Da Cunha Rodrigues et al., 2016), (Qu & Buyya, 2014), (Almutiry, Wills, Alwabel, Crowder, & Walters, 1998), (Kara et al., 2017), (Alkhanak et al., 2015), (Pawluk, Litoiu, & Cercone, 2011)
1.2.5 Scalability	Medida de la capacidad del sistema de una aplicación, para proporcionar mayor rendimiento, menor tiempo de respuesta y/o dar soporte a más usuarios cuando se agregan recursos de hardware sin modificaciones en forma rentable. Se refiere a la capacidad del servicio en la nube suministrado por el CSP, para manejar el rango de solicitudes realizadas simultáneamente por las aplicaciones. La capacidad de adaptarse a la demanda adicional de recursos también es una parte esencial de la flexibilidad requerida por el proveedor de computación en la nube. Se distinguen dos tipos de escalabilidad: vertical y horizontal. <ul style="list-style-type: none"> • Escalabilidad vertical: la ampliación vertical o ampliación significa agregar más recursos a un nodo de plataforma determinado, como los núcleos de la CPU o la memoria, de manera que el nodo de la plataforma pueda manejar una carga de trabajo mayor. 	(Baranwal & Vidyarthi, 2014), (Grandhi & Wibowo, 2016), (Abdeladim et al., 2014), (Ghahramani et al., 2017), (Abuhussein et al., 2016), (Zhang, Zhou, Li, & Gao, 2017), (H., M. Khan, Chan, & Chua, 2016), (Mrabet et al., 2017b), (G. Raj et al., 2014), (Monir et al., 2015), (Medeiros et al., 2017), (Sebastian Lehrig et al., 2015), (Da Cunha Rodrigues et al., 2016), (Nadanam & Rajmohan, 2012), (J. Y. Lee et al., 2009), (Klems et al., 2012), (Baliyan & Kumar, 2013), (L. Lu & Yuan, 2018), (Muñoz-Escóí & Bernabéu-Aubán, 2017), (Baranwal & Vidyarthi, 2016)

Atributo	Definición	Trabajos relacionados
	<ul style="list-style-type: none"> Escalabilidad horizontal: escalar horizontalmente o escalar significa agregar nuevos nodos a un clúster o sistema distribuido de manera que todo el sistema pueda manejar grandes cargas de trabajo. <p>Consideraciones incluidas: 1) ¿El CSP garantiza la disponibilidad al ampliar hacia arriba/abajo? 2) ¿El CSP permite escalar a diferentes zonas geográficas?</p>	
1.2.5.1 Customizability	Se define como el número de cambios dinámicos en un servicio, sobre el número de cambios dinámicos en el servicio más el número de cambios estáticos en el servicio. Todo esto respecto a una carga de trabajo.	(S. Singh & Chana, 2015)
1.2.5.2 Scalability range	Métrica refleja la capacidad de un sistema de computación en la nube para lograr sus objetivos de nivel de servicio en un cierto rango de carga de trabajo, por ejemplo, un rango de tasas de solicitud. Para cada carga de trabajo individual dentro de este rango, el sistema logra sus SLO. El rango de carga de trabajo se define como una carga de trabajo máxima.	(Becker et al., 2015), (S. Lehrig & Becker, 2015)
1.2.5.3 Scalability speed	Rango de carga de trabajo con una tasa de cambio máxima en la que un sistema puede escalar. Es una métrica de escalabilidad que considera la velocidad a la que un sistema puede escalar. Es decir, la métrica define que un sistema puede lograr sus SLO en cada momento en que la carga de trabajo cambia a una tasa de cambio máxima. La tasa se define por una carga de trabajo máxima y una tasa de aumento.	(Becker et al., 2015), (S. Lehrig & Becker, 2015)
1.2.5.4 Virtual infrastructure automation	Escalado automático, implementación y configuración.	(De Benedetti et al., 2015)
1.2.5.5 Virtualization technique	Generalmente un proveedor admite solo una plataforma de virtualización. Cada técnica de virtualización tiene sus propias ventajas. Si alguien no requiere características avanzadas, puede buscar Hyper-V. Xen está basado en Linux y su gestión y administración es complicada, sin embargo, presenta ventajas	(Baranwal & Vidyarthi, 2014), (Brataas, Herbst, Ivansek, & Polutnik, 2017), (Xiao & Xiao, 2013), (Baranwal & Vidyarthi, 2016), (H. Lee, Kim, Lee, & Won, 2012)

Atributo	Definición	Trabajos relacionados
	para usuarios que tienen experiencia en linux. VMware proporciona características robustas. Existen otros proveedores que ofrecen Vsphere. Hyper-V se distribuye sin costo, con restricciones de copyright. La elección de una técnica de virtualización depende de los requerimientos de la aplicación a desplegar.	

5.4.3. Característica “1.3 Assurance”

1.3	Assurance	6
1.3.1	Availability	11
1.3.1.1	Down Time	
1.3.1.2	Failure Rate	
1.3.1.3	Jitter	
1.3.1.4	Latency	
1.3.1.5	Response time	
1.3.1.6	Service Time	
1.3.1.7	Service use rate	
1.3.1.8	Successability	
1.3.1.9	Time Efficiency	
1.3.1.10	Up Time	
1.3.1.11	Waiting Time	
1.3.2	Dependability	
1.3.3	Maintainability	
1.3.4	Recoverability	
1.3.5	Reliability	10
1.3.5.1	Completeness	
1.3.5.2	Continuity	
1.3.5.3	Correctness	
1.3.5.4	Coverage of Failure Recovery	
1.3.5.5	Coverage of Fault Tolerance	
1.3.5.6	Packet Loss	
1.3.5.7	Punctuality	
1.3.5.8	Reputation	
1.3.5.9	Reusability	
1.3.5.10	Stability	
1.3.6	Resilience	

Esta categoría ha sido abordada en cuatro trabajos: (M. Z. Khan & Qamar, 2015), (Shin & Huh, 2015), (Siegel & Perdue, 2012), (Yadav & Goraya, 2018)

Esta categoría incluye atributos clave que indican la probabilidad de que el servicio esté disponible según lo especificado.

Los atributos relacionados con la característica Assurance son descritos en la Tabla 27.

Tabla 27. Atributos de “Assurance”

Atributo	Definición	Trabajos relacionados
1.3.1 Availability	<p>Es el grado en que un sistema o componente es operativo y accesible cuando es requerido.</p> <p>Es el porcentaje de tiempo en que un recurso se encuentra disponible cuando un cliente lo requiere.</p> <p>Al evaluar la disponibilidad se debe tener en cuenta: (1) ¿El proveedor del servicio asegura la disponibilidad para todos los clientes en una máquina física? (2) ¿El proveedor protege las máquinas virtuales que comparten una partición ante ataques de tipo DoS?</p>	<p>(Baranwal & Vidyarthi, 2014), (De Benedetti et al., 2015), (Yaqub et al., 2014), (Zhou et al., 2015), (Z. Tan et al., 2016), (Ghahramani et al., 2017), (Abuhussein et al., 2016), (Hongzhen, Limin, Dehua, & Yanqin, 2016), (Gholami & Arani, 2015), (Lim & Thiran, 2014), (H. M. Khan et al., 2016), (Mrabet et al., 2017a), (X. Zheng et al., 2014), (Mrabet et al., 2017b), (Elizabeth, Ramya, Prakash, & Uthariaraj, 2014), (Monir et al., 2015), (Techio & Misaghi, 2015), (Wagle et al., 2016), (Jelassi, Ghazel, & Saidane, 2017), (Bruneo, 2014), (Ritu & Jain, 2017), (Undheim, Chilwan, & Heegaard, 2011), (Karim, Ding, & Miri, 2013), (H. Liu, Bu, & Cai, 2012), (Nadanam & Rajmohan, 2012), (Motta, You, Sacco, & Sfondrini, 2013), (Salama et al., 2012), (J. Y. Lee et al., 2009), (Khaddaj, 2012), (Jhawar & Piuri, 2013), (Ghosh & Ghosh, 2012), (Tao, Chang, Yi, & Gu, 2010), (Klems et al., 2012), (Baliyan & Kumar, 2013), (M. Ahmed et al., 2014), (Alkhanak et al., 2015), (Lin, Lai, Wu, & Lo, 2014), (Nabi, Toeroe, & Khendek, 2016), (S. Singh & Chana, 2015), (L. Lu & Yuan, 2018), (Halabi & Bellaiche, 2018), (Abderrahim & Choukair, 2018), (Deshpande, Sharma, Peddoju, & Abraham, 2018), (Z.-H. Tan, Liu, Shi, Wang, & Guo, 2017), (R. R. Kumar & Kumar, 2017), (Cai, Zhang, Zhou, Zhao, & Li, 2017), (Baranwal & Vidyarthi, 2016), (Mohamed, 2016), (Tchernykh et al., 2016), (Shu, Zhao, Liu, Zuo, & Yang, 2015), (Aljazzaf, 2015), (Sousa, Moreira, Santos, & Machado, 2012), (Z. E. Wang & Liu, 2012)</p>
1.3.1.1 Down time	<p>Tiempo durante el cual los servicios no están disponibles. Este tiempo debe ser minimizado por el proveedor de servicios y ante la necesidad de una actualización, reparación o mantenimiento debe existir una planificación</p>	<p>(M. K. Naseer et al., 2014)</p>

Atributo	Definición	Trabajos relacionados
	propuesta por el proveedor.	
1.3.1.2 Failure rate	Promedio de fallas de una máquina virtual en una hora.	(Qu & Buyya, 2014)
1.3.1.3 Jitter	Variación en el retraso de los paquetes recibidos. La unidad de medida es en milisegundos.	(Jelassi et al., 2017), (Pedersen et al., 2011)
1.3.1.4 Latency	Es la diferencia de tiempo entre el momento en que un bit de un paquete entra y sale de un canal. Depende de varios factores como: longitud del paquete, latencia de la red y rendimiento de la red del proveedor.	(Ghahramani et al., 2017), (Jelassi et al., 2017), (Ritu & Jain, 2017), (M. Ahmed et al., 2014), (Chen, Chang, Tseng, Huang, & Lei, 2011), (S. Singh & Chana, 2015), (Deshpande et al., 2018), (Aljazzaf, 2015)
1.3.1.5 Response time	Es el tiempo entre la solicitud de un servicio y la disponibilidad de la respuesta por parte del proveedor. Este tiempo depende de la infraestructura y de la aplicación para la cual se ha generado la solicitud. También se suele considerar como el promedio y el tiempo máximo que tarda un proveedor en responder, según lo acordado en los acuerdos de nivel de servicio. La unidad de medida es milisegundos.	(Baranwal & Vidyarthi, 2014), (Grandhi & Wibowo, 2016), (Tang & Yan, 2015), (Zhang, Han, Li, Leung, & Song, 2016), (Hongzhen et al., 2016), (Lim & Thiran, 2014), (H. M. Khan et al., 2016), (Elizabeth et al., 2014), (M. K. Naseer et al., 2014), (Monir et al., 2015), (Jelassi et al., 2017), (Grati, Boukadi, & Ben-Abdallah, 2014), (Karim et al., 2013), (H. Liu et al., 2012), (Nadanam & Rajmohan, 2012), (Salama et al., 2012), (Tao et al., 2010), (Klems et al., 2012), (M. Ahmed et al., 2014), (Lin et al., 2014), (Ré et al., 2018), (L. Lu & Yuan, 2018), (R. R. Kumar & Kumar, 2017), (Baranwal & Vidyarthi, 2016), (Shu et al., 2015), (Aljazzaf, 2015), (Sousa et al., 2012), (Z. E. Wang & Liu, 2012)
1.3.1.6 Service time	Tiempo necesario para ejecutar una máquina virtual. Puede ser calculada como el tiempo requerido para ejecutar una máquina virtual de manera local y el tiempo necesario para ejecutar una máquina virtual en una en una nube federada.	(Bruneo, 2014)
1.3.1.7 Service use rate	Razón entre el tiempo del servicio en la nube y el tiempo final utilizado.	(H. M. Khan et al., 2016), (Ritu & Jain, 2017), (S. Singh & Chana, 2015), (Aljazzaf, 2015)
1.3.1.8 Successability	Número de respuestas dividido por número de peticiones.	(M. Ahmed et al., 2014)

Atributo	Definición	Trabajos relacionados
1.3.1.9 Time efficiency	El tiempo efectivo es aquel en el cual se da respuesta a una solicitud dentro de los tiempos de respuesta establecidos en los acuerdos de nivel de servicio. Es la proporción del número de veces que se completan las solicitudes, dentro de los tiempos de respuesta acordados entre cliente y proveedor.	(Mrabet et al., 2017b)
1.3.1.10 Uptime	Tiempo máximo en el cual los servicios están disponibles por parte del proveedor. La eficiencia de un proveedor de servicio en la nube se puede determinar a partir del tiempo en el cual los servicios están activos a lo largo de un período de tiempo.	(M. K. Naseer et al., 2014)
1.3.1.11 Waiting time	Es el tiempo que espera un usuario en cola por una respuesta.	(Bruneo, 2014), (Ran, Yang, Zhang, & Xi, 2017), (Alkhanak et al., 2015)
1.3.2 Dependability	Es la medida en cual el servicio es confiable bajo condiciones definidas en un acuerdo de nivel de servicio.	(Monir et al., 2015), (Salama et al., 2012)
1.3.3 Maintainability	Capacidad del proveedor de servicios en la nube para realizar modificaciones en el servicio, manteniendo el servicio en buenas condiciones operativas.	(Pandey & Daniel, 2016), (White et al., 2017), (Abderrahim & Choukair, 2018), (Shin & Huh, 2016), (Choi & Jeong, 2014)
1.3.4 Recoverability	Es el grado en que un servicio puede reanudar rápidamente un estado normal de operación después de una interrupción no planificada. Mide la preparación del proveedor para recuperar y proteger el servicio ante un desastre. Entre las consideraciones principales están: (1) ¿El proveedor del servicio ofrece un plan de recuperación de desastres? (2) ¿Está automatizado el proceso de recuperación?	(Abuhussein et al., 2016), (H. Liu et al., 2012), (Song et al., 2012), (Ghosh & Ghosh, 2012)
1.3.5 Reliability	La confiabilidad se define como el grado en	(Baranwal & Vidyarthi, 2014), (De Benedetti et al., 2015), (M.

Atributo	Definición	Trabajos relacionados
	<p>que un servicio en la nube está disponible, completo, correcto, continuo, estable y consistente. Refleja las medidas de cómo funciona un servicio sin fallas, bajo determinadas condiciones durante un período de tiempo. También se considera como la tasa de éxito de las tareas ejecutadas.</p> <p>Entre las consideraciones principales están:</p> <p>(1) ¿El proveedor del servicio verifica regularmente las condiciones de seguridad y privacidad para los servicios ofrecidos? (2) ¿Tiene el proveedor del servicio una política clara de subcontratación?</p>	<p>Naseer & Nazar, 2017), (Zhou et al., 2015), (Z. Tan et al., 2016), (Ghahramani et al., 2017), (Abuhussein et al., 2016), (Hongzhen et al., 2016), (Gholami & Arani, 2015), (Kim et al., 2014), (Lim & Thiran, 2014), (Zhang et al., 2017), (Mrabet et al., 2017a), (X. Zheng et al., 2014), (Mrabet et al., 2017b), (Elizabeth et al., 2014), (Monir et al., 2015), (Pandey & Daniel, 2016), (Techio & Misaghi, 2015), (Jelassi et al., 2017), (Grati et al., 2014), (Ritu & Jain, 2017), (Z. Wang et al., 2015), (Karim et al., 2013), (Faragardi, Shojaee, Tabani, & Rajabi, 2013), (Song et al., 2012), (Nadanam & Rajmohan, 2012), (J. Y. Lee et al., 2009), (Khaddaj, 2012), (Sodhi & Prabhakar, 2012), (Tao et al., 2010), (Klems et al., 2012), (M. Ahmed et al., 2014), (W. Ahmed & Wu, 2013), (Alabool et al., 2018), (Alkhanak et al., 2015), (S. Singh & Chana, 2015), (White et al., 2017), (L. Lu & Yuan, 2018), (Abderrahim & Choukair, 2018), (Ma et al., 2018), (Deshpande et al., 2018), (Z.-H. Tan et al., 2017), (R. R. Kumar & Kumar, 2017), (Baranwal & Vidyarathi, 2016), (Shin & Huh, 2016), (Aljazzaf, 2015), (Choi & Jeong, 2014), (Sivaraj & Muniraj, 2014), (Z. E. Wang & Liu, 2012)</p>
1.3.5.1 Completeness	Grado en que un servicio en la nube soporta todas las tareas y objetivos especificados por el cliente.	(Zhou et al., 2015), (Oh et al., 2011), (Nadanam & Rajmohan, 2012), (Almutiry et al., 1998), (Pawluk et al., 2011),
1.3.5.2 Continuity	Grado en que se ofrece un servicio en la nube bajo todas las circunstancias, incluye la mitigación de riesgos y la recuperación de los servicios después de una suspensión.	(Zhou et al., 2015), (Mrabet et al., 2017b), (Tao et al., 2010)
1.3.5.3 Correctness	Grado en que un servicio en la nube utiliza los procesos correctos y produce los resultados correctos. Mide la preparación del servicio con respuestas y acciones preconfiguradas para enfrentar las fallas de seguridad y privacidad. Esto incluye detección automática, prevención y respuesta ante fallas. Entre las	(Zhou et al., 2015), (Abuhussein et al., 2016), (Zhang et al., 2017), (S. Singh & Chana, 2015)

Atributo	Definición	Trabajos relacionados
	<p>consideraciones principales están:</p> <p>1) ¿El proveedor del servicio realiza acciones para identificar las causas que ocasionan un mal servicio?</p> <p>2) ¿El proveedor del servicio tiene herramientas para prevenir automáticamente los riesgos de seguridad y privacidad?</p>	
1.3.5.4 Coverage of failure recovery	Mide la proporción de fallas que son corregidas en un período de tiempo específico.	(Padmapriya & Rajmohan, 2012)
1.3.5.5 Coverage of fault tolerance	Es un mecanismo de garantía de calidad de servicio con tolerancia a fallos, que busca mejorar la capacidad del proveedor, para superar fallas.	(Padmapriya & Rajmohan, 2012), (Freitas, Parlavantzas, & Pazat, 2012)
1.3.5.6 Packet loss	Es el porcentaje de paquetes enviados o recibidos que no llegan a su destino, con respecto al número total de paquetes transmitidos durante un intervalo de tiempo específico.	(Jelassi et al., 2017)
1.3.5.7 Punctuality	Hace referencia a la capacidad de un proveedor para cumplir requerimientos en un periodo de tiempo.	(Mrabet et al., 2017b)
1.3.5.8 Reputation	Mide la confiabilidad de un proveedor de la nube y se basa en la experiencia de los usuarios con los proveedores de servicios.	(Baranwal & Vidyarthi, 2014), (Werner et al., 2015), (Tang & Yan, 2015), (Hongzhen et al., 2016), (Jelassi et al., 2017), (Karim et al., 2013), (Salama et al., 2012), (Tao et al., 2010), (Ma et al., 2018), (Bilecki, Fiorese, & Matos, 2017), (Baranwal & Vidyarthi, 2016)
1.3.5.9 Reusability	El principio para definir este atributo se explica por la visión del consumidor y del proveedor del servicio. Normalmente los proveedores de servicios web en la nube utilizan un modelo de negocio de pago por uso. Si el servicio web ofrece funcionalidades que cumplen con los	(Nadanam & Rajmohan, 2012), (J. Y. Lee et al., 2009), (Baliyan & Kumar, 2013)

Atributo	Definición	Trabajos relacionados
	<p>requisitos de los clientes, esto permitirá aumentar el número de clientes de servicios y el retorno de la inversión (ROI). Este atributo evalúa si las funcionalidades proporcionadas por los servicios son comunes a los requisitos definidos por los clientes. Por lo tanto, la reutilización se considera como un indicador que puede revelar el ROI de los proveedores de servicios.</p>	
1.3.5.10 Stability	<p>Es la capacidad de un servicio en la nube para estar continua y constantemente disponible.</p>	<p>(Zhou et al., 2015), (Tang & Yan, 2015), (Mrabet et al., 2017b), (Tao et al., 2010), (White et al., 2017)</p>
1.3.6 Resilience	<p>Es la capacidad de un sistema que está expuesto a peligros para resistir y recuperarse de manera oportuna y eficiente, incluida la preservación y restauración de sus estructuras y funciones básicas esenciales. Un sistema resistente es aquel que muestra probabilidades de fallas reducidas, consecuencias mínimas de las fallas y tiempo de recuperación corto.</p> <p>Los proveedores de servicio utilizan estrategias como conexiones redundantes y duplicación de recursos o información.</p>	<p>(Ghahramani et al., 2017), (H. M. Khan et al., 2016)</p>

5.4.4. Característica “1.4 Empathy”

1.4	Empathy	5
1.4.1	Courtesy	
1.4.2	Initiative	
1.4.3	Measurability	
1.4.4	Satisfaction	
1.4.5	Self-service	

Esta categoría ha sido abordada en tres trabajos: (M. Naseer & Nazar, 2017), (Zhou et al., 2015), (Z. Wang et al., 2015)

Se fundamenta en las experiencias de los clientes con los proveedores. Empatía se define como el grado en que el proveedor atiende los requerimientos de un cliente de manera cortés.

Los atributos relacionados con la característica Empathy son descritos en la Tabla 28.

Tabla 28. Atributos de “Empathy”

Atributo	Definición	Trabajos relacionados
1.4.1 Courtesy	Grado en que un servicio es provisto de manera educada, respetuosa y amigable.	(Zhou et al., 2015)
1.4.2 Initiative	Grado en que el proveedor del servicio en la nube reconoce los requisitos de los clientes y sugiere cambios para satisfacer sus necesidades.	(Zhou et al., 2015)
1.4.3 Measurability	Capacidad que permite que un servicio en la nube se pueda monitorear, controlar, reportar y facturar.	(Zhou et al., 2015)
1.4.4 Satisfaction	Juicio cualitativo o cuantitativo que permite saber si un servicio proporcionó un nivel placentero de cumplimiento conforme a lo esperado.	(Kim et al., 2014), (Tao et al., 2010),
1.4.5 Self-service	Grado en el que un cliente puede aprovisionar y administrar unilateralmente los recursos de la nube según sea necesario.	(Zhou et al., 2015), (Monir et al., 2015)

5.4.5. Característica “1.5 Finacial”

1.5	Financial	2
1.5.1	Billing process	
1.5.2	Cost	4
1.5.2.1	Data Transfer Cost	
1.5.2.2	Liquidated damages	
1.5.2.3	Storage Cost	
1.5.2.4	VM Cost	

Esta categoría ha sido abordada en cuatro trabajos: (M. Z. Khan & Qamar, 2015), (Shin & Huh, 2015), (Siegel & Perdue, 2012), (Yadav & Goraya, 2018)

Hace referencia a la cantidad de dinero invertido en el servicio por parte del cliente. El costo tiende a ser la métrica más cuantificable hoy en día en los servicios en la nube. El costo de los servicios refleja el modelo de negocios de la organización.

Los atributos relacionados con la característica Finacial son descritos en la Tabla 29.

Tabla 29. Atributos de “Financial”

Atributo	Definición	Trabajos relacionados
1.5.1 Billing process	Nivel de integración entre el cliente y los sistemas de facturación del proveedor con el fin de llevar un control de las facturas periódicas.	(S. Singh & Chana, 2015)
1.5.2 Cost	Valor impuesto por el proveedor a un servicio, incluye el valor del servicio y los posibles valores adicionales por el uso de este.	(Baranwal & Vidyarthi, 2014), (Grandhi & Wibowo, 2016), (Tang & Yan, 2015), (Brataas et al., 2017), (Jelassi et al., 2017), (Grati et al., 2014), (Ritu & Jain, 2017), (Ran et al., 2017), (Karim et al., 2013), (Tao et al., 2010), (Salama et al., 2012), (Alkhanak et al., 2015), (Lin et al., 2014), (R. R. Kumar & Kumar, 2017), (Baranwal & Vidyarthi, 2016), (Z. E. Wang & Liu, 2012), (Pawluk et al., 2011)
1.5.2.1 Data transfer cost	Costo de transferir datos dentro o fuera del centro de datos.	(Qu & Buyya, 2014)
1.5.2.2 Liquidated damage	Es la compensación que el proveedor del servicio tiene con un cliente ante un fallo o incumplimiento del acuerdo de nivel de servicio.	(Z. E. Wang & Liu, 2012)
1.5.2.3 Storage cost	Costo del almacenamiento secundario, se mide en gigas por mes.	(Qu & Buyya, 2014)
1.5.2.4 VM cost	Es el costo de utilizar una máquina virtual, se mide en horas por mes.	(Qu & Buyya, 2014)

5.4.6. Característica “1.6 Performance”

1.6	Performance	9
1.6.1	Accuracy	
1.6.2	Capacity	5
1.6.2.1	CPU capacity metric	1
1.6.2.1.1	CPU speed	
1.6.2.2	CPU occupancy rate	
1.6.2.3	Memory capacity metric	2
1.6.2.3.1	Memory Read	
1.6.2.3.2	Memory Write	
1.6.2.4	Network capacity metric	3
1.6.2.4.1	Inbound	
1.6.2.4.2	Outbound	
1.6.2.4.3	Task integrity	
1.6.2.5	Storage capacity metric	2
1.6.2.5.1	Disk Read	
1.6.2.5.2	Disk Write	
1.6.3	Data transfer	
1.6.4	Functionality	
1.6.5	Interoperability	
1.6.6	Specific Service	
1.6.7	Suitability	1
1.6.7.1	Dedicated hardware	
1.6.8	Throughput and Efficiency	2
1.6.8.1	Marginal costs	
1.6.8.2	Resource Provisioning Efficiency	
1.6.9	Utilization	

Esta categoría ha sido abordada en diecisiete trabajos: (De Benedetti et al., 2015), (M. Z. Khan & Qamar, 2015), (Yaqub et al., 2014), (Zhang et al., 2017), (Shin & Huh, 2015), (G. Raj et al., 2014), (Medeiros et al., 2017), (Siegel & Perdue, 2012), (Motta et al., 2013), (Salama et al., 2012), (Khaddaj, 2012), (Freitas et al., 2012), (Cao et al., 2018), (S. Singh & Chana, 2015), (White et al., 2017), (Yadav & Goraya, 2018), (Mohamed, 2016)

Cubre las características y funciones de los servicios prestados. Los proveedores de servicios ofrecen múltiples soluciones que satisfacen las necesidades de TI de diferentes organizaciones. Cada solución tiene un rendimiento diferente en términos de funcionalidad, tiempo de respuesta del servicio, precisión, entre otros.

Los atributos relacionados con la característica Performance son descritos en la Tabla 30.

Tabla 30. Atributos de "Performance"

Atributo	Definición	Trabajos relacionados
1.6.1 Accuracy	Es la medida en que un servicio cumple totalmente con los requisitos del cliente.	(Padmapriya & Rajmohan, 2012), (Nadanam & Rajmohan, 2012), (Almutiry et al., 1998), (Salleh et al., 2017), (L. Lu & Yuan, 2018), (Shin & Huh, 2016), (Pawluk et al., 2011)
1.6.2 Capacity	Es la cantidad máxima de recursos que un proveedor de servicios puede proporcionar en las horas pico.	(Baranwal & Vidyarthi, 2014), (Brataas et al., 2017), (Sebastian Lehrig et al., 2015), (Z. Y. Wang et al., 2012), (Baranwal & Vidyarthi, 2016)
1.6.2.1 CPU capacity metric	Es la medida que se obtiene al multiplicar el número de CPUs y la frecuencia de ellas.	(Baranwal & Vidyarthi, 2014), (Ré et al., 2018), (Phani Sheetal & Ravindranath, 2018)
1.6.2.1.1 CPU speed	Número máximo de instrucciones ejecutadas por un solo núcleo en unidad de tiempo.	(Qu & Buyya, 2014)
1.6.2.2 CPU occupancy rate	Hace referencia a la proporción de instancias de servicio que requieren capacidad de cómputo. La sumatoria de las instancias representan el uso total de la capacidad de computación total disponible.	(Hongzhen et al., 2016)
1.6.2.3 Memory capacity metric	Es una medida de almacenamiento cuya unidad es en gigas.	(Baranwal & Vidyarthi, 2014), (Ré et al., 2018), (Phani Sheetal & Ravindranath, 2018)
1.6.2.3.1 Memory read	Cantidad máxima de datos transferidos desde la memoria en una unidad de tiempo, su unidad es megabits por segundo.	(Qu & Buyya, 2014)
1.6.2.3.2 Memory write	Cantidad máxima de datos escritos en la memoria en una unidad de tiempo, su unidad de medida es megabits por segundo.	(Qu & Buyya, 2014)
1.6.2.4 Network capacity metric	Medida relacionada con el ancho de banda que suministra el proveedor del servicio.	(Baranwal & Vidyarthi, 2014), (Isa, Brogi, & Forti, 2016), (Ritu & Jain, 2017), (S. Singh & Chana, 2015), (Phani Sheetal & Ravindranath, 2018)
1.6.2.4.1 Inbound	Cantidad máxima de datos transferidos a una máquina virtual en una unidad de tiempo, su medida es en gigabits por segundo.	(Qu & Buyya, 2014)

Atributo	Definición	Trabajos relacionados
1.6.2.4.2 Outbound	Cantidad máxima de datos transferidos fuera de la máquina virtual en una unidad de tiempo, su unidad de medida es gigabits por segundo.	(Qu & Buyya, 2014)
1.6.2.4.3 Task integrity	Nivel de completitud de una tarea que inicia y termina sin pérdidas por afectaciones de red u otros factores.	(Z.-H. Tan et al., 2017)
1.6.2.5 Storage capacity metric	Medida de almacenamiento de datos, usualmente se mide en gigas.	(Baranwal & Vidyarthi, 2014)
1.6.2.5.1 Disk read	Cantidad máxima de datos transferidos desde el almacenamiento secundario en una unidad de tiempo (Mb/s).	(Qu & Buyya, 2014)
1.6.2.5.2 Disk write	Cantidad máxima de datos escritos en el almacenamiento secundario en una unidad de tiempo (Mb/s).	(Qu & Buyya, 2014)
1.6.3 Data transfer	Mide la rapidez con la que se proporciona la transferencia de datos en el servicio suministrado.	(Ghahramani et al., 2017)
1.6.4 Functionality	Hace referencia a las características específicas proporcionadas por un servicio.	(Shin & Huh, 2016), (Choi & Jeong, 2014)
1.6.5 Interoperability	Capacidad de un servicio para interactuar fácilmente con otros servicios ya sea con servicios del mismo proveedor o de otros.	(De Benedetti et al., 2015), (Medeiros et al., 2017), (Salleh et al., 2017), (White et al., 2017), (L. Lu & Yuan, 2018), (Olokunde et al., 2017), (Assis & Bittencourt, 2016), (Shin & Huh, 2016)
1.6.6 Specific service	Los servicios específicos están relacionados directamente con el proveedor del servicio. Generalmente estos servicios son únicos o tienen características particulares que los diferencian de servicios ofrecidos por otros proveedores. Los servicios específicos sirven para dar reconocimiento a un proveedor.	(M. K. Naseer et al., 2014)
1.6.7 Suitability	Refleja el grado en que el proveedor de computación en la nube cumple con los requisitos de la organización para ofrecer servicios o aplicaciones	(Grandhi & Wibowo, 2016), (Salleh et al., 2017), (Shin & Huh, 2016)

Atributo	Definición	Trabajos relacionados
	específicas. El proveedor de la nube debe satisfacer los requisitos críticos y no críticos de la organización.	
1.6.7.1 Dedicated hardware	<p>Mide la disponibilidad del servicio para hacer frente a los problemas de disponibilidad que resultan de compartir recursos. Entre las consideraciones principales están:</p> <ol style="list-style-type: none"> 1) ¿El proveedor del servicio ofrece máquinas dedicadas a un solo cliente? 2) ¿El proveedor del servicio aplica en máquinas dedicadas las mismas contramedidas de seguridad que se aplican a las máquinas virtuales? 	(Abuhussein et al., 2016)
1.6.8 Throughput an efficiency	El rendimiento hace referencia a la cantidad de tareas completadas por el proveedor de servicios en una unidad de tiempo. Una solicitud está formada por una colección de tareas. El rendimiento de una aplicación depende del proveedor de servicios y del funcionamiento de la aplicación.	(Baranwal & Vidyarthi, 2014), (H. M. Khan et al., 2016), (Jelassi et al., 2017), (Grati et al., 2014), (Sebastian Lehrig et al., 2015), (H. Liu et al., 2012), (Pedersen et al., 2011), (Nadanam & Rajmohan, 2012), (Salama et al., 2012), (J. Y. Lee et al., 2009), (Klems et al., 2012), (Tolosana-Calasanz, Bañares, Pham, & Rana, 2012), (M. Ahmed et al., 2014), (Lin et al., 2014), (L. Lu & Yuan, 2018), (Ma et al., 2018), (R. R. Kumar & Kumar, 2017), (Baranwal & Vidyarthi, 2016), (Shin & Huh, 2016), (Aljazzaf, 2015), (Choi & Jeong, 2014), (Sivaraj & Muniraj, 2014), (Sousa et al., 2012), (Becker et al., 2015)
1.6.8.1 Marginal cost	Los costos marginales son los costos de operación en los que incurre una unidad de carga de trabajo adicional, esto permite medir la eficiencia de un sistema desplegado en la nube y cuantificarlo para obtener su costo.	(Becker et al., 2015), (S. Lehrig & Becker, 2015)
1.6.8.2 Resource Provisioning efficiency	Se define la eficiencia de aprovisionamiento de recursos como una métrica para medir la eficiencia de un sistema en un delta de carga de trabajo específico. La métrica mide el desajuste entre la utilización real de recursos y la demanda de recursos mientras la carga	(Becker et al., 2015), (S. Lehrig & Becker, 2015)

Atributo	Definición	Trabajos relacionados
	de trabajo está cambiando. Un sistema perfectamente eficiente adaptará sus recursos asignados a la demanda actual de la aplicación.	
1.6.9 Utilization	Es la relación entre la cantidad de recursos físicos utilizados y la cantidad total de recursos físicos.	(Bruneo, 2014)

5.4.7. Característica “1.7 Security and Privacy”

1.7	Security and privacy	12
1.7.1	Access Control	4
1.7.1.1	Attribute based Encryption	
1.7.1.2	Authentication and Authorization	
1.7.1.3	Identity Management and Authentication	
1.7.1.4	Revocation	
1.7.2	Backup	
1.7.3	Confidentiality	
1.7.4	Data Security	8
1.7.4.1	Data accuracy	
1.7.4.2	Data Availability	
1.7.4.3	Data Confidentiality	
1.7.4.4	Data consistency	
1.7.4.5	Data correctness	
1.7.4.6	Data geographic /political	
1.7.4.7	Data Integrity	
1.7.4.8	Data Sanitization	
1.7.5	Network security support	
1.7.6	Obligations	
1.7.7	Physical and environmental security	
1.7.8	Proactive threat and vulnerability management	
1.7.9	Risks	
1.7.10	Safety	
1.7.11	Security management	
1.7.12	Traceability	

Esta categoría ha sido abordada en treinta y tres trabajos: (De Benedetti et al., 2015), (M. Naseer & Nazar, 2017), (M. Z. Khan & Qamar, 2015), (Zhou et al., 2015), (Kim et al., 2014), (Zhang et al., 2017), (X. Zheng et al., 2014), (Shin & Huh, 2015), (G. Raj et al., 2014), (Monir et al., 2015), (Medeiros et al., 2017), (Techio & Misaghi, 2015), (Qu & Buyya, 2014), (Z. Wang et al., 2015), (Siegel & Perdue, 2012), (Karim et al., 2013), (Song et al., 2012), (Khaddaj, 2012), (Sodhi & Prabhakar, 2012), (Tao et al., 2010), (Alkhanak et al., 2015), (Chang, Ramachandran, Yao, Kuo, & Li, 2016), (Salleh et al., 2017), (S. Singh & Chana, 2015), (White et al., 2017), (Yadav & Goraya, 2018), (Ma et al., 2018), (R. R. Kumar & Kumar, 2017), (Baranwal & Vidyarthi, 2016), (Tchernykh et al., 2016), (Shin & Huh, 2016), (Z. E. Wang & Liu, 2012)

En esta categoría se incluyen los atributos que indican la efectividad de los controles de los proveedores sobre el acceso a los servicios, los datos del servicio y las instalaciones físicas sobre las que se ofrecen los servicios.

El grado en que un servicio en la nube protege los activos de los clientes y el acceso a su información, así como el acceso a los datos para que los clientes u otros sistemas tengan el un adecuado de acceso de acuerdo con sus niveles de autorización.

Los atributos relacionados con la característica Security and Privacy son descritos en la Tabla 31.

Tabla 31. Atributos de “Security and Privacy”

Atributo	Definición	Trabajos relacionados
1.7.1 Access control	<p>Un sistema de control de acceso incluye componentes y métodos para especificar políticas de seguridad para los usuarios legítimos. Entre las consideraciones principales están:</p> <p>1) ¿El proveedor de servicios proporciona una separación entre los niveles de acceso para un administrador y un usuario sin privilegios?</p> <p>2) ¿El proveedor de servicios permite el control total de las listas de control de acceso?</p>	(Rajeswari & Kalaiselvi, 2018), (Abuhussein et al., 2016), (G. Raj et al., 2014), (Chang et al., 2016), (Ma et al., 2018)
1.7.1.1 Attribute based Encryption	<p>Los datos de cada usuario utilizan un cifrado basado en atributos (ABE) para su protección. Esto permite que solo usuarios autorizados puedan acceder a su información. El proveedor debe suministrar un método de protección para los datos que llegan y salen del servidor, de tal manera que no puedan ser manipulados sin la clave de cifrado respectiva. Entre las consideraciones principales están:</p> <p>1) ¿Los datos transferidos desde y hacia la nube están cifrados por defecto?</p> <p>2) ¿Cumple el atributo de cifrado FIPS 140-2?</p>	(Baranwal & Vidyarthi, 2014), (Rajeswari & Kalaiselvi, 2018), (Abuhussein et al., 2016)
1.7.1.2 Authentication and Authorization	<p>La autenticación y autorización (A&A) es un desafío importante en la próxima generación de Internet (NGI). A&A se enfoca en dos niveles: a nivel de usuario dentro de la red y a nivel de aplicación en una plataforma informática. El objetivo principal de A&A es que los usuarios se desplacen entre redes sin inconvenientes, independientemente que sean cableadas o inalámbricas. La autenticación debe suministrar permiso para acceder a las aplicaciones y la autorización debe proveer los permisos necesarios para que un usuario solo pueda acceder a aquello que tiene permiso.</p>	(Talukder & Prahalad, 2009), (Ghosh & Ghosh, 2012), (Assis & Bittencourt, 2016)
1.7.1.3 Identity	<p>Hace referencia a la solidez del servicio contra el acceso no</p>	(Abuhussein et al., 2016), (Monir et al.,

Atributo	Definición	Trabajos relacionados
Management and Authentication	<p>autorizado a los componentes, mediante la verificación de las credenciales de inicio de sesión. Entre las consideraciones principales están:</p> <p>1) ¿Se admite la autenticación multifactor (MFA)?</p> <p>2) ¿Está el sistema de autenticación sincronizado con la organización del directorio activo?</p>	2015)
1.7.1.4 Revocation	<p>Medios para invalidar la información utilizada por la autenticación y la autorización de forma dinámica. Los usuarios pueden transferir las credenciales para realizar la autenticación y de esta manera liberar los atributos de seguridad entre los diferentes proveedores de la nube. La revocación permite garantizar la suspensión inmediata de solicitudes de autenticación y autorización.</p>	(Werner et al., 2015)
1.7.2 Backup	<p>Capacidad de archivar código y datos de una aplicación en momentos específicos. Mide la preparación del servicio para responder ante fallas, pérdidas o daños en los datos almacenados, haciendo copias de seguridad. Entre las consideraciones principales están:</p> <p>1) ¿El ancho de banda de respaldo/restauración garantiza la transferencia de los datos?</p> <p>2) ¿Está automatizado el servicio de copia de seguridad?</p>	(Yaqub et al., 2014), (Abuhussein et al., 2016)
1.7.3 Confidentiality	<p>Capacidad de un servicio en la nube para garantizar que los datos son accesibles solo para aquellos usuarios que pueden tener acceso. El proveedor debe suministrar políticas y procesos para garantizar que solo el personal con los privilegios apropiados pueda hacer uso o modificar los datos producto del trabajo y uso de las aplicaciones.</p>	(Zhou et al., 2015), (Talukder & Prahalad, 2009), (Ghosh & Ghosh, 2012), (Tchernykh et al., 2016)
1.7.4 Data Security	<p>Medida que permite calificar los mecanismos de seguridad que protegen los datos del usuario.</p>	(Qu & Buyya, 2014)
1.7.4.1 Data accuracy	<p>Hace referencia al resultado observado cuando los valores de los datos almacenados para un objeto son correctos. Los datos deben tener una única representación de tal manera que eviten la ambigüedad.</p>	(Zhang et al., 2017)

Atributo	Definición	Trabajos relacionados
1.7.4.2 Data Availability	Es la capacidad de un proveedor para ofrecer productos y servicios que aseguren que los datos continúen disponibles cuando sean requeridos en situaciones normales o extremas. Una estrategia para garantizar la disponibilidad de los datos es la redundancia.	(Moghaddam, Majd, Ahmadi, Khodadadi, & Madadipouya, 2016)
1.7.4.3 Data Confidentiality	Es responsabilidad del proveedor garantizar la protección de la información ante accesos no autorizados, por tanto, las restricciones de uso e intercambio de datos entre los clientes están reguladas por las políticas del proveedor. Las fallas o accesos no autorizados deben ser detectadas y reportadas a los clientes.	(Moghaddam et al., 2016), (Karim et al., 2013), (Ghosh & Ghosh, 2012)
1.7.4.4 Data consistency	Hace referencia a los métodos de recopilación de datos y a la consistencia de los mismos en diferentes ubicaciones. Es útil para evaluar la consistencia de grandes conjuntos de datos desde perspectivas diferentes.	(Zhang et al., 2017), (Almutiry et al., 1998), (Klems et al., 2012), (S. Singh & Chana, 2015), (Sousa et al., 2012)
1.7.4.5 Data correctness	Es un parámetro de calidad para los datos, sirve para evaluar la corrección de grandes volúmenes de datos en términos de tipos de datos y formatos.	(Zhang et al., 2017)
1.7.4.6 Data geographic /political	Identifica las limitaciones por parte del cliente para almacenar la información en función de la ubicación del servicio y el posible riesgo geográfico o político. El proveedor del servicio debe proteger sus centros de datos tanto física como lógicamente. Entre las consideraciones principales están: 1) ¿El proveedor del servicio oculta al público las ubicaciones exactas de los centros de datos? 2) ¿Pueden los clientes elegir la ubicación geográfica donde almacenar sus datos?	(Ghahramani et al., 2017), (Abuhussein et al., 2016), (Monir et al., 2015), (Z. Y. Wang et al., 2012)
1.7.4.7 Data Integrity	Capacidad que tiene un servicio en la nube para evitar el acceso no autorizado a los datos. El proveedor debe mantener los datos almacenados en forma correcta, para que los clientes puedan estar seguros de que su información es precisa y válida.	(Zhou et al., 2015), (Z. Tan et al., 2016), (Rajeswari & Kalaiselvi, 2018), (Abuhussein et al., 2016), (Gholami & Arani, 2015), (Zhang et al., 2017), (Mrabet et al., 2017a), (Mrabet et al., 2017b), (Elizabeth et al., 2014), (Monir et al., 2015), (Moghaddam et al., 2016), (Medeiros et al., 2017), (Khaddaj, 2012), (Ghosh & Ghosh, 2012),

Atributo	Definición	Trabajos relacionados
		(Chang et al., 2016), (S. Singh & Chana, 2015), (Halabi & Bellaiche, 2018), (Ma et al., 2018), (Aloraini & Hammoudeh, 2017), (Tchernykh et al., 2016)
1.7.4.8 Data Sanitization	<p>Identifica las medidas de seguridad que toma un proveedor de servicios para eliminar o destruir datos de forma deliberada, permanente e irreversible (en medios de almacenamiento, documentos, etc.) para que no puedan recuperarse. Entre las consideraciones principales están:</p> <p>1) ¿Se destruyen los datos de forma segura cuando se termina el servicio?</p> <p>2) ¿El proveedor de servicios desinfecta los medios de almacenamiento antes de ser reutilizados?</p>	(Abuhussein et al., 2016)
1.7.5 Network security support	<p>Identifica la efectividad del proveedor del servicio para prevenir el acceso no autorizado, el uso indebido, la modificación o la denegación de redes virtuales y físicas en la nube. Entre las consideraciones principales están:</p> <p>1) ¿El proveedor permite a los consumidores construir redes geográficamente dispersas?</p> <p>2) ¿El proveedor proporciona contramedidas para los ataques a la red?</p>	(Baranwal & Vidyarthi, 2014), (Abuhussein et al., 2016), (Qu & Buyya, 2014)
1.7.6 Obligations	<p>Acciones que deben realizarse cuando ocurren ciertos eventos. Las obligaciones son esenciales en la gestión de credenciales y sus atributos para cumplir con la legislación o incluso los acuerdos de nivel de servicio establecidos, según un perfil o un contexto de gestión de identidad.</p>	(Werner et al., 2015)
1.7.7 Physical and environmental security	<p>Políticas y procesos en uso por parte del proveedor de servicios, para proteger las instalaciones ante un acceso físico no autorizado, daños o interferencias. Mide la capacidad del proveedor para proporcionar conocimiento a todas las partes interesadas, en temas relacionados con la protección de los activos físicos e información. Entre las consideraciones principales están:</p>	(Abuhussein et al., 2016), (Qu & Buyya, 2014)

Atributo	Definición	Trabajos relacionados
	1) ¿El proveedor proporciona recomendaciones para una mejor seguridad? 2) ¿El proveedor advierte a los consumidores sobre posibles vulnerabilidades?	
1.7.8 Proactive threat and vulnerability management	El proveedor debe implementar mecanismos para garantizar que el servicio esté protegido contra las amenazas recurrentes conocidas y contra las nuevas vulnerabilidades en desarrollo. Mide la solidez del proceso para garantizar que la máquina virtual es segura. Esto incluye sus cálculos, redes y almacenamiento. Entre las consideraciones principales están: 1) ¿El proveedor evita la implementación de máquinas virtuales inseguras o manipuladas? 2) ¿Puede el proveedor identificarse y defenderse contra un ataque a las máquinas virtuales?	(Abuhussein et al., 2016)
1.7.9 Risks	El usuario debe tener la autonomía, aun conociendo los riesgos, para elegir un servidor o servicio en particular y tener la confianza para liberar sus datos en ellos. Evalúa la bondad del proveedor al identificar, evaluar y priorizar, monitorear y controlar los riesgos. Las consideraciones incluyen: 1) ¿El proveedor tiene un plan de administración de riesgos para los servicios ofrecidos? 2) ¿El proveedor permite la participación de auditores externos certificados en todas las actividades de gestión de riesgos?	(Werner et al., 2015), (Abuhussein et al., 2016)
1.7.10 Safety	El proveedor tiene a su disposición tecnología que permite asegurar los servicios.	(Tang & Yan, 2015), (Abderrahim & Choukair, 2018)
1.7.11 Security management	Capacidad del proveedor de servicios para garantizar la seguridad de la aplicación, los datos y la infraestructura en función de los requisitos de seguridad del cliente. Entre las consideraciones principales están: 1) ¿El proveedor promueve protocolos seguros como SSL o IPsec? 2) ¿El proveedor hace el mejor esfuerzo para aumentar la conciencia en cuanto a seguridad y privacidad en los clientes?	(Abuhussein et al., 2016), (M. K. Naseer et al., 2014), (Qu & Buyya, 2014)

Atributo	Definición	Trabajos relacionados
1.7.12 Traceability	La capacidad del proveedor para permitir que las actividades y procesos relacionadas con los servicios puedan ser registradas y rastreadas.	(Zhou et al., 2015)

5.4.8. Característica “1.8 Usability”

1.8	Usability	17
1.8.1	Accessibility	
1.8.2	Appropriateness Recognizability	
1.8.3	Changeability	
1.8.4	Client Interface	
1.8.5	Documentation	
1.8.6	Free Trial	
1.8.7	Group Formation	
1.8.8	Learnability	1
1.8.8.1	API	
1.8.9	Operability	
1.8.10	Revision	
1.8.11	Share	
1.8.12	Synchronization	
1.8.13	Tangibility	2
1.8.13.1	Professionalism	
1.8.13.2	Visibility	
1.8.14	Transparency	
1.8.15	Understandability	
1.8.16	User error protection	
1.8.17	Version Control	

Esta categoría ha sido abordada en dieciséis trabajos: (M. Naseer & Nazar, 2017), (M. Z. Khan & Qamar, 2015), (Zhou et al., 2015), (X. Zheng et al., 2014), (Shin & Huh, 2015), (Pandey & Daniel, 2016), (Tajvidi, Ranjan, Kolodziej, & Wang, 2014), (Siegel & Perdue, 2012), (Karim et al., 2013), (Almutiry et al., 1998), (S. Singh & Chana, 2015), (White et al., 2017), (Yadav & Goraya, 2018), (Olokunde et al., 2017), (Shin & Huh, 2016), (Choi & Jeong, 2014)

El grado en que un servicio en la nube puede ser utilizado por usuarios diferentes para lograr objetivos específicos con efectividad, eficiencia y satisfacción.

Es la facilidad con la que se puede utilizar un servicio. Además, describe qué tan fácil, eficiente y agradable es utilizar la interfaz para un servicio, y permite evaluar la facilidad de invocación del servicio cuando se expone como una Interfaz de programación de aplicaciones (API).

Los atributos relacionados con la característica Usability son descritos en la Tabla 32.

Tabla 32. Atributos de “Usability”

Atributo	Definición	Trabajos relacionados
1.8.1 Accessibility	Es la facilidad con la que se puede acceder o extraer información de un sistema.	(Kim et al., 2014), (Khaddaj, 2012), (Lin et al., 2014), (Aljazzaf, 2015)
1.8.2 Appropriateness Recognizability	Grado en que los usuarios pueden reconocer si un servicio es apropiado para sus necesidades.	(Zhou et al., 2015), (Kim et al., 2014)
1.8.3 Changeability	Se puede definir como la sumatoria de las variables: Tiempo para analizar el cambio en la carga de trabajo, Tiempo para modificar el cambio en la carga de trabajo, Tiempo para probar el cambio en la carga de trabajo y tiempo para distribuir el cambio en la carga de trabajo, todas ellas sobre el número de solicitudes de cambio en la carga de trabajo.	(S. Singh & Chana, 2015)
1.8.4 Client Interface	Grado en que la interfaz del servicio permite una interacción agradable y satisfactoria para el cliente.	(Baranwal & Vidyarathi, 2014), (Zhou et al., 2015), (Baranwal & Vidyarathi, 2016)
1.8.5 Documentation	Cantidad y calidad de la documentación que un proveedor ofrece sobre sus servicios.	(M. Ahmed et al., 2014)
1.8.6 Free Trial	Capacidad de los proveedores para ofrecer pruebas gratuitas para probar sus servicios. Para los clientes es un factor primordial debido a que permite probar los servicios antes de hacer una implementación con ellos.	(Baranwal & Vidyarathi, 2014), (S. Singh & Chana, 2015), (Baranwal & Vidyarathi, 2016)
1.8.7 Group Formation	Facilidad con la cual se pueden crear grupos, asignar miembros a los grupos e identificarlos.	(Kim et al., 2014)
1.8.8 Learnability	Grado en que los usuarios pueden aprender a utilizar un servicio para lograr un nivel específico de efectividad y eficiencia, en una cantidad determinada de tiempo. Se puede medir como el esfuerzo que requiere un usuario para aprender a utilizar el servicio.	(Zhou et al., 2015)
1.8.8.1 API	Indica la disposición del proveedor para ofrecer interfaces de programación de aplicaciones API que permitan interactuar con el servidor.	(Baranwal & Vidyarathi, 2014), (Baranwal & Vidyarathi, 2016)
1.8.9 Operability	Grado en que un servicio en la nube tiene atributos que facilitan su operación y control. Es la capacidad de un servicio para ser fácilmente operado por los	(Zhou et al., 2015), (Kim et al., 2014), (Techio & Misaghi, 2015)

Atributo	Definición	Trabajos relacionados
	usuarios.	
1.8.10 Revision	Capacidad del proveedor para ofrecer un seguimiento del registro de eventos por parte de una persona con permisos.	(Kim et al., 2014)
1.8.11 Share	Capacidad del proveedor para permitir que una aplicación sea compartida entre varias personas.	(Kim et al., 2014)
1.8.12 Synchronization	Facilidad ofrecida por el proveedor para ajustar un sistema y permitir que funcione con otros de manera sincronizada.	(Kim et al., 2014)
1.8.13 Tangibility	Grado en que un servicio ofrecido por un proveedor permite la comunicación con otros servicios de manera eficiente.	(M. Naseer & Nazar, 2017), (Zhou et al., 2015), (Z. Wang et al., 2015)
1.8.13.1 Professionalism	Grado en el cual influye la experiencia de los empleados para que los usuarios tengan una mejor usabilidad de los servicios.	(Zhou et al., 2015)
1.8.13.2 Visibility	Grado en que los clientes potenciales conocen el servicio de un proveedor.	(Zhou et al., 2015), (Song et al., 2012), (Assis & Bittencourt, 2016)
1.8.14 Transparency	Medida en que los usuarios pueden determinar cuándo ocurren los cambios en una característica o componente del servicio y si estos cambios afectan la usabilidad. La transparencia es una propiedad de un sistema que es capaz de "dar cuenta", o proporcionar visibilidad, de cómo se ajusta a sus reglas y compromisos.	(G. Raj et al., 2014), (Baliyan & Kumar, 2013), (Nuñez et al., 2016)
1.8.15 Understandability	Facilidad con la que los usuarios pueden comprender las capacidades y el funcionamiento de los servicios.	(Alabool et al., 2018)
1.8.16 User error protection	Capacidad de un servicio para orientar a los usuarios a que no cometan errores.	(Zhou et al., 2015)
1.8.17 Version Control	Capacidad del proveedor para permitir un control sobre la concurrencia que evite el conflicto de versiones.	(Kim et al., 2014)

5.4.9. Característica “1.9 User experience”

1.9	📁 User Experience	1
1.9.1	📄 Cloud Customer Confidence Level	

Esta categoría ha sido abordada en tres trabajos: (Baranwal & Vidyarthi, 2014), (Ma et al., 2018), (Baranwal & Vidyarthi, 2016)

Un usuario interesado en utilizar un servicio puede considerar opiniones de otros usuarios con respecto a la precisión, estabilidad y la transparencia servicio.

Los atributos relacionados con la característica User experience son descritos en la Tabla 33.

Tabla 33. Atributos de “User experience”

Atributo	Definición	Trabajos relacionados
1.9.1 Cloud Customer Confidence Level	Es la razón entre el número de cargas de trabajo exitosas sobre el número total de cargas de trabajo enviadas.	(S. Singh & Chana, 2015)

6. MODELO DE CALIDAD

Un modelo de calidad se define como una representación de las perspectivas de calidad, las relaciones entre las características, subcaracterísticas y atributos de calidad asociados a una entidad, los cuales permiten su evaluación a partir de métricas (IGI-Global, 2018). De acuerdo con J. Y. Lee et al. (2009) y Olokunde, Misra, & Adewumi (2017), un modelo de calidad puede desarrollarse en las siguientes fases: 1) definición de características claves del área de estudio, 2) determinación de atributos de calidad de las características claves, 3) definición de métricas para cada atributo de calidad y 4) evaluación del modelo.

Los modelos propuestos por Wen & Dong (2013), X. Zheng, Martin, Brohman, & Xu (2014), Zhou, Wang, Li, & Jiang (2015) y Techio & Misaghi (2015), presentados en detalle en la sección “2 TRABAJOS RELACIONADOS”, coinciden en la definición de dimensiones o perspectivas para la clasificación de las características de calidad, este aspecto se tomó en cuenta para definir el modelo de calidad. Ahora bien, la clasificación de las características y atributos de calidad del modelo propuesto toma como referencia la TQACA.

6.1. Alcance

De acuerdo con los objetivos específicos definidos en esta investigación, el modelo de calidad propuesto busca proveer un instrumento que pueda ser utilizado por desarrolladores en la fase de requerimientos. Al ser una extensión de la taxonomía TQACA realizada en la presente investigación, reutiliza la totalidad de las características y atributos clasificados en 3 vistas: desarrollador, proveedor y usuario. Esto le permitirá a un desarrollador, seleccionar los atributos de calidad que puede tener en cuenta en la fase de requerimientos, en el marco del desarrollo de una aplicación que va a ser desplegada en la nube.

6.2. El nuevo modelo y sus vistas

La taxonomía desarrollada en la presente investigación fue la base para la construcción del modelo. TQACA permitió: 1) definir las características claves que permiten agrupar todos los atributos de calidad. 2) determinar la relación de los atributos de calidad, así como su ubicación en una estructura jerárquica y 3) definir las métricas para cada atributo de calidad.

Las características claves del modelo de calidad propuesto, corresponden a las 9 categorías de primer nivel de la taxonomía. La Figura 23 presenta el primer nivel de TQACA.

1.1	▷	📁	Accountability
1.2	▷	📁	Agility
1.3	▷	📁	Assurance
1.4	▷	📁	Empathy
1.5	▷	📁	Financial
1.6	▷	📁	Performance
1.7	▷	📁	Security and privacy
1.8	▷	📁	Usability
1.9	▷	📁	User Experience

Figura 23. Características claves del modelo de calidad – Primer nivel de la taxonomía

TQACA para cada atributo, define clasificaciones en 5 contextos: tipo, unidad, alcance, categoría y tipo de requerimiento. La Figura 24 presenta a manera de ejemplo, la información de cada contexto para el atributo *Contracting experience*.

The screenshot displays a taxonomy tree on the left with columns for Order, Categories / Attributes, and Nodes. The 'Contracting experience' attribute (1.1.2) is selected. On the right, the configuration panel shows the following settings:

- Type:** Quantitative
- Unit:** Ordered set
- Scope:** User
- Category:** Subjective
- Requirement:** Non-functional requirement

Figura 24. Tipos de clasificación para atributos

Con base en el contexto “alcance” y soportado en la garantía literaria que permitió definir la taxonomía, se definieron tres vistas para el nuevo modelo: vista usuario, vista desarrollador y vista proveedor. La Figura 25 presenta las vistas del modelo propuesto y las características claves definidas en detalle en TQACA.

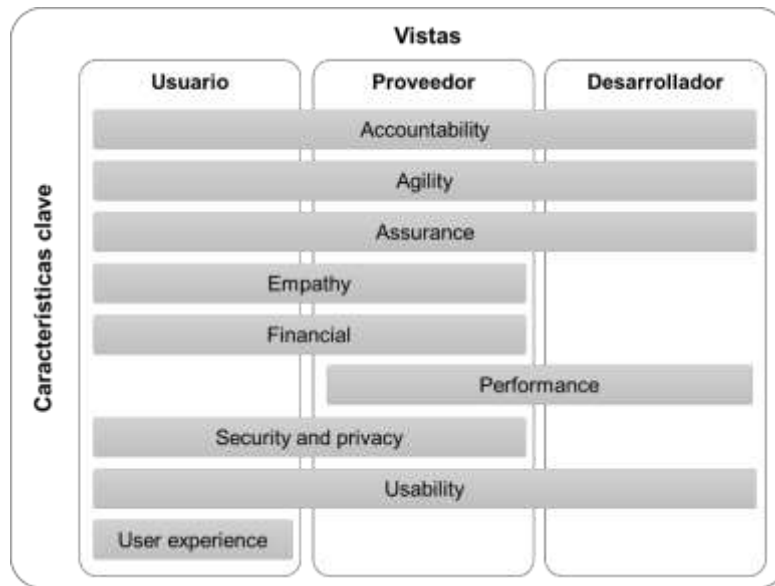


Figura 25. Modelo de calidad TQACA

Como se puede apreciar en la Figura 25 no todas las características claves definidas en TQACA son transversales a las vistas del modelo. La transversalidad se da por la aparición de por lo menos un atributo de calidad en toda la estructura jerárquica de TQACA.

En los siguientes apartados se presentarán las vistas y la clasificación de los atributos en cada característica clave.

6.2.1. Vista de usuario

Esta vista contiene 8 de las 9 características claves de TQACA. La Figura 26 presenta los atributos de calidad que hacen parte de la vista de usuario, sin embargo, no contiene todos los atributos que se encuentran en la taxonomía. Esto se debe a que la organización de las vistas está soportada en el contexto “alcance” de cada uno de los atributos, el cual está fundamentado en la garantía literaria.

Los atributos de calidad con lineado intermedio representan las características que no hacen parte de la vista, sin embargo, estas características pueden tener algún atributo de calidad en su interior que si pertenezca. Por ejemplo, se puede apreciar el atributo “Accountability”, esto indica que este atributo no pertenece directamente a esta vista, no obstante, tiene 4 atributos que si pertenecen a la vista usuario. Esta particularidad del modelo se presenta porque es una extensión de TQACA y mantiene total coherencia con la fuente primaria.

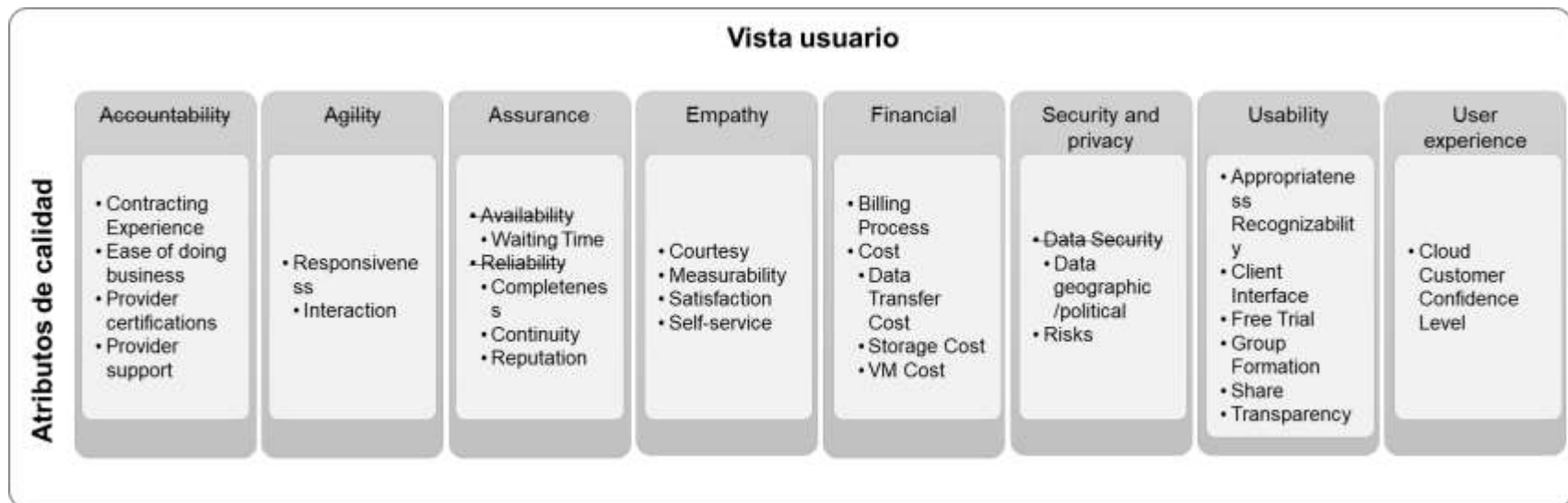


Figura 26. Vista usuario

6.2.2. Vista de proveedor

La vista proveedor presenta 8 características de TQACA y agrupa los atributos de calidad de acuerdo con su estructura jerárquica. En la Figura 27, se puede apreciar que los atributos “Empathy” y “Finacial” no pertenecen a esta vista, sin embargo, algunos atributos de calidad de su la estructura jerárquica definida en TQACA si hacen parte.

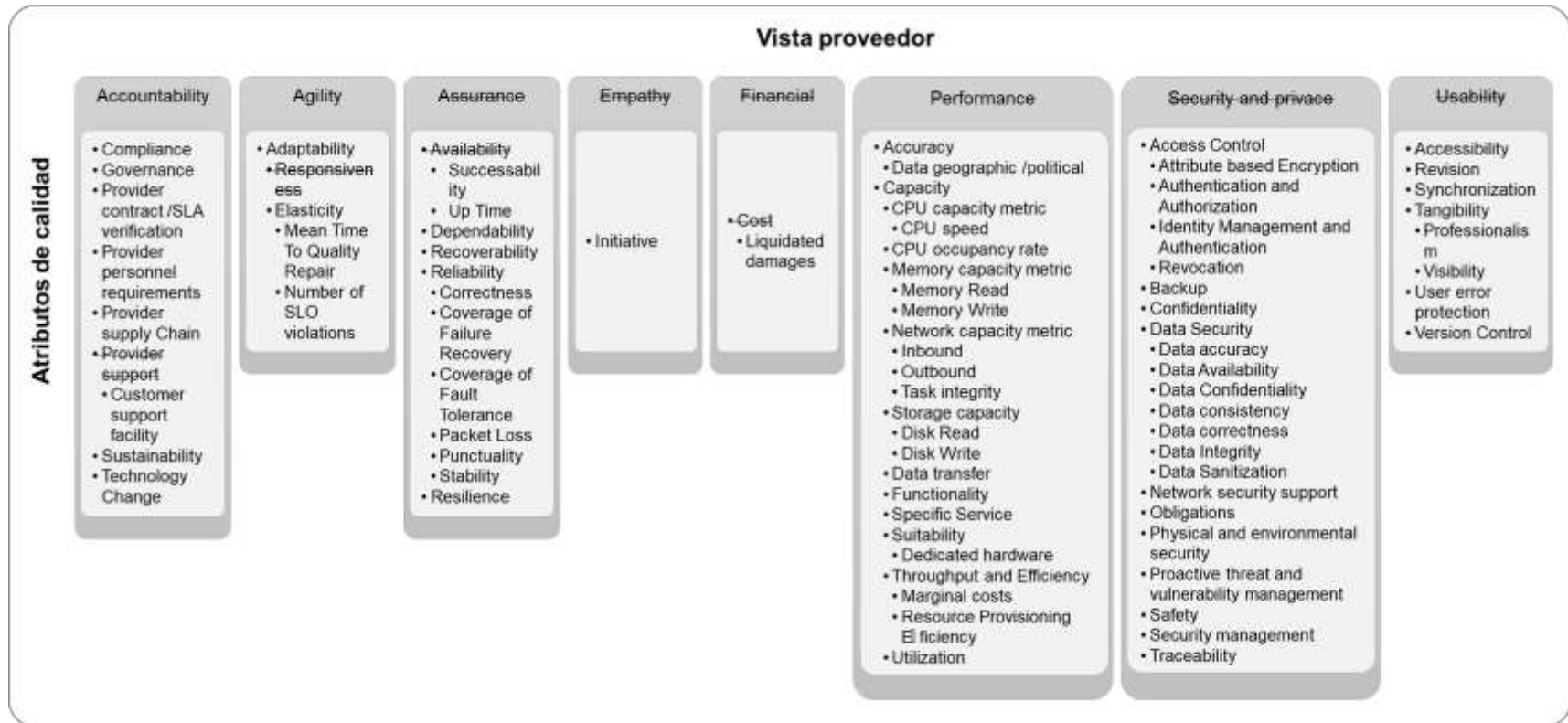


Figura 27. Vista proveedor

6.2.3. Vista desarrollador

La vista del desarrollador contiene 5 de las 9 características de TQACA. La representación conserva la misma metodología de las vistas anteriores como se aprecia en la Figura 28.

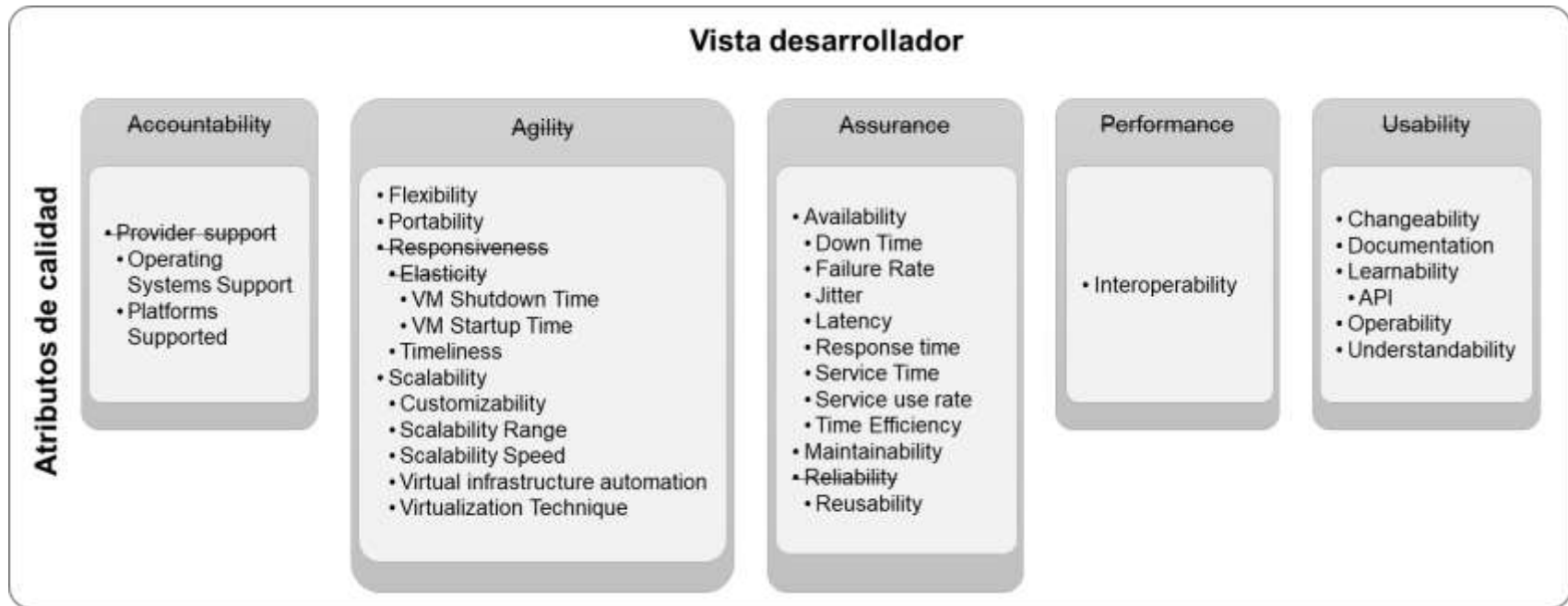


Figura 28. Vista desarrollador

6.3. Instrumento para la validación del modelo de calidad

6.3.1. Contexto industria de software

Para la validación del modelo se diseñó un instrumento, que permitió la verificación de los 121 atributos de calidad identificados. El proceso de calibración del instrumento generó tres versiones.

La primera versión del instrumento presentaba 4 columnas en un documento de Excel. En la primera columna se encontraba el nombre del atributo de calidad, en la segunda columna la definición encontrada en la literatura, en la tercera columna la métrica asociada al atributo y en la cuarta columna, una lista de opciones para identificar con qué frecuencia, el desarrollador tenía presente el uso del atributo, al realizar el despliegue de una aplicación en la nube. Este instrumento fue calibrado por 4 desarrolladores con experiencia en el desarrollo de aplicaciones para la nube. Los hallazgos encontrados fueron: 1) Alto nivel de complejidad en las definiciones de los atributos. 2) Se requería un perfil superior a desarrollador para su entendimiento y 3) El instrumento era muy extenso.

Teniendo en cuenta lo anterior, se diseñó una segunda versión del instrumento. Se mantuvo el nombre del atributo de calidad y la lista de opciones. Para cada atributo se definió una pregunta que le permitiera al desarrollador comprender el contexto de uso del atributo. Esta pregunta fue formulada con base en la definición y métrica del atributo encontrada en TQACA. Este instrumento fue calibrado por 6 desarrolladores. Los hallazgos encontrados fueron: 1) Oportunidades de mejora en la redacción de algunas preguntas y 2) El instrumento era muy extenso. Con base en los hallazgos encontrados en la calibración de la segunda versión del instrumento, se realizaron los ajustes de redacción sugeridos por los desarrolladores. No obstante, el tamaño del instrumento no fue posible ajustarlo debido a que los 121 atributos debían ser evaluados. La Figura 29 presenta una muestra de la versión en español del instrumento.

TQACA Extended Quality Model			
Modelo de Calidad Extendido para la Taxonomía de Atributos de Calidad en Aplicaciones para la Nube			
A continuación, encontrará un conjunto de preguntas que tienen como objetivo, conocer la frecuencia con la cual usted tiene en cuenta los atributos de calidad al momento de realizar el despliegue de una aplicación en la nube.			
Atributo	Pregunta	Frecuencia	Comentarios sobre su selección (Opcional)
Compliance	¿Usted se asegura que el proveedor de la nube cumpla cabalmente con los estándares definidos?	Escoja una opción	
Contracting experience	¿Usted tiene en cuenta experiencias anteriores para cambiar el proveedor en la nube?	Escoja una opción	
Ease of doing business	¿Verifica los niveles de satisfacción de otros desarrolladores o usuarios para cambiar el proveedor?	Escoja una opción	
Governance	¿Usted despliega sus aplicaciones solo en proveedores altamente reconocidos?	Escoja una opción	
Provider certifications	¿Usted tiene en cuenta las certificaciones que tiene el proveedor en la nube para ofrecer sus servicios?	Escoja una opción	
Provider contract / SLA verification	¿Usted verifica el contrato de nivel de servicio establecido, antes de desplegar su aplicación?	Escoja una opción	
Provider personnel requirements	¿Usted verifica que el proveedor en la nube tenga personal de soporte calificado para atender sus requerimientos?	Escoja una opción	
Provider supply chain	¿Usted tiene en cuenta que el proveedor tenga contratos con terceros para ofrecerle servicios?	Escoja una opción	

...Los atributos van hasta el 121

Figura 29. Instrumento validación del modelo

La lista de opciones utilizada en el instrumento final de validación presentaba cuatro alternativas (columna frecuencia). La Tabla 34 presenta los textos utilizados y su relación cuantitativa en la investigación.

Tabla 34. Lista de opciones instrumento validación

Valor cualitativo	Valor cuantitativo
Nunca	0
Algunas veces	1
Usualmente	2
Siempre	3

6.3.2. Contexto academia

El instrumento diseñado y calibrado para el contexto industria de software, fue utilizado en la prueba piloto con dos grupos de estudiantes universitarios. Como condición inicial para participar en el piloto se estableció que los estudiantes debían haber cursado y aprobado la asignatura calidad de software o una asignatura con temática similar.

Durante la prueba piloto, se evidenció que la mayor parte de los estudiantes no conocían los atributos de calidad presentados en el instrumento, por lo tanto, el instrumento fue nuevamente calibrado para este contexto. Se incluyó la opción “No reconocido” a la lista de opciones, permitiendo dar respuesta a cada una de las preguntas del instrumento. La Tabla 35 presenta la nueva lista de opciones utilizada en el instrumento.

Tabla 35. Lista de opciones instrumento validación – contexto academia

Valor cualitativo	Valor cuantitativo
No reconocido	N/A
Nunca	0
Algunas veces	1
Usualmente	2
Siempre	3

6.4. Selección de la muestra

6.4.1. Contexto industria de software

Para la selección de la muestra se utilizó el muestreo no probabilístico discrecional (Medianero-Burga, 2011), estableciendo como criterios de selección:

- ✓ Desarrolladores de software con experiencia en despliegue de aplicaciones en la nube.
- ✓ Desarrolladores vinculados directa o indirectamente a una empresa que desarrolla software para la nube.
- ✓ Puntos de cobertura donde se estima existen individuos que pueden hacer parte de la muestra.

El tipo de muestreo utilizado no permite realizar aseveraciones sobre el riesgo, margen de error y el nivel de confianza. Esto se debe al sesgo que puede existir en la definición de los criterios de selección de la muestra. No obstante, con el fin de presentar resultados cuantitativos sobre el trabajo realizado y tomando como referencia trabajos similares (Meyer, Ramirez, & Pérez, 2013), (Miguel, Alessandro, & David, 2016), (Mantecón, Juan, Calafat, Becoña, & Román, 2008), se asume que la muestra es representativa para el universo de estudio. La Figura 30 presenta la fórmula utilizada para el cálculo del tamaño de la muestra.

$$n = \frac{Z_{\alpha}^2 \cdot p \cdot q}{i^2}$$

Figura 30. Cálculo de la muestra con tamaño de población desconocida

Se asignó un valor porcentual a la población “*p*”, estableciendo el valor más pesimista. Por su parte, “*q*” representa el complemento de la proporción muestral conocida. “*Z*” corresponde al nivel de confianza de la muestra, se tomó el valor correspondiente a la tabla muestral para *Z*[95]. Para el porcentaje de error “*i*” se estableció un valor soportado en: 1) El estudio no representa un riesgo para los seres humanos y 2) La precisión de los resultados podría determinar la inclusión o exclusión de un atributo de calidad en la guía final para sugerir atributos de calidad en la fase de requerimientos y/o diseño, sin embargo, la totalidad de los atributos están contemplados tanto en el modelo propuesto como en la taxonomía, por lo tanto, no es un riesgo para la validez de este estudio. La Tabla 36 presenta los valores utilizados en la investigación.

Tabla 36. Valores variables

Variable	Valor
<i>p</i>	0.5
<i>q</i>	(1-05) = 0.5
<i>Z</i>	<i>Z</i> [95] = 1.96
<i>i</i>	15%
<i>n</i>	43

Por lo anterior, se determinó un tamaño mínimo de la muestra de 43 desarrolladores para la aplicación del instrumento.

6.4.2. Contexto academia

Como se mencionó en la sección “1.2 Objetivos de la investigación”, la investigación esperaba analizar el impacto de la taxonomía en proyectos de desarrollo en los contextos: industria de software y academia. Para la selección de la muestra se aplicó el mismo tipo

de muestreo que para el contexto industria de software, es decir, muestreo no probabilístico discrecional. Los criterios para la selección de la muestra fueron:

- ✓ Estudiantes de instituciones de educación superior con acreditación institucional, de programas de sistemas o informática en metodología presencial.
- ✓ Estudiantes de último semestre o en su defecto que hubiesen cursado y aprobado asignaturas relacionadas con la calidad del software.

Con el fin de conocer la cantidad de universidades que cumplían con el primer criterio de selección, se consultó el Sistema Nacional de Información de la Educación Superior (SNIES) teniendo como datos de búsqueda las instituciones acreditadas a nivel institucional. Posteriormente se consultaron los programas de ingeniería de sistemas, con metodología presencial de las instituciones acreditadas encontradas. Se evidenciaron 18 programas que cumplían con las características definidas en los criterios de selección. Se seleccionaron 5 grupos de estudiantes de 3 programas académicos para participar en la validación del modelo.

6.5. Aplicación del instrumento

6.5.1. Contexto industria de software

Para coordinar la aplicación del instrumento, se establecieron 3 puntos de acceso (hotspot). La Figura 31 presenta los puntos de acceso definidos.



Figura 31. Puntos de acceso para coordinar aplicación instrumento

El primer hotspot se ubicó en Bogotá Colombia, coordinado directamente por el investigador principal. El segundo hotspot coordinado por un investigador experto en desarrollo de aplicaciones, ubicado en Roswell, Georgia. El tercer hotspot coordinado por una empresa de desarrollo con sede en Colombia y desarrolladores en la India. La Tabla 37 presenta los valores de la aplicación del instrumento por cada hotspot.

Tabla 37. Aplicación del instrumento contexto industria

País	Planificado	Real
Colombia	14	24
Estados Unidos	14	12
India	15	16
Total	43	52

6.5.2. Contexto academia

El instrumento de validación del modelo fue aplicado en 3 universidades con acreditación institucional, que ofrecen el programa de Ingeniería de Sistemas. La Tabla 38 presenta la muestra sobre la cual se aplicó el instrumento.

Tabla 38. Aplicación del instrumento contexto academia

Grupos	Planificado	Real
Grupo 1	8	7
Grupo 2	22	20
Grupo 3	14	14
Grupo 4	6	6
Grupo 5	21	21

6.6. Definición de elementos claves para el análisis de resultados

6.6.1. Análisis por realizar

La aplicación del instrumento busca el establecimiento de un ranking de atributos de calidad para cada una de las vistas del modelo. El ranking permitirá definir cuáles son los atributos que tienen en cuenta con mayor frecuencia los desarrolladores en cada una de las vistas del modelo.

6.6.2. Objetivo del análisis

A partir del ranking definido, se busca establecer cuartiles para cada una de las vistas del modelo. Cada cuartil permitirá agrupar los atributos de calidad de acuerdo con una frecuencia de uso por parte de los desarrolladores.

6.6.3. Justificación

El modelo busca proporcionar instrumentos que permitan a los desarrolladores, seleccionar atributos de calidad que deberían ser tenidos en cuenta al momento de definir una especificación de requerimientos para una aplicación a ser desplegada en la nube. La clasificación de los atributos en cuartiles proporcionará a los desarrolladores una guía sobre los atributos de calidad más utilizados por sus colegas. Esta clasificación presentará todos los atributos, organizados por cuartiles, a partir de la frecuencia de uso que se obtenga con la aplicación de los instrumentos.

6.7. Resultados de la aplicación del instrumento

6.7.1. Contexto industria de software

Para cada atributo de calidad, se realizó la sumatoria de las respuestas suministradas por cada desarrollador. El modelo de calidad definido en esta investigación propone 3 vistas, por lo tanto, los resultados de la aplicación del instrumento serán presentados de esta forma.

6.7.1.1. Vista desarrollador

De los 121 atributos del modelo, 28 corresponden a la vista del desarrollador. La Tabla 39 presenta el ranking de los resultados obtenidos.

Tabla 39. Resultados vista desarrollador

Atributo	Sumatoria
Down time	133
Scalability speed	131
Virtual infrastructure automation	123
Flexibility	122
Maintainability	119
Time efficiency	118
Operability	117
API	116
Service time	115
Operating system support	114
VM Startup time	114
Changeability	112
Platforms supported	109
Portability	108
Latency	108
Customizability	107
Timeliness	104
Scalability range	103
VM Shutdown time	101
Understandability	101
Interoperability	94
Failure rate	92
Service use rate	87

Atributo	Sumatoria
Jitter	83
Reusability	80
Response time	79
Virtualization technique	76
Documentation	76

Para la vista desarrollador, el atributo de calidad más relevante fue *Down Time*, y el de menos importancia fue *Documentation*. El ranking permite utilizar medidas de posición para clasificar la información en cuartiles. Cada cuartil representa los valores del 25, 50 y 75 por ciento de los datos que son menores o iguales a un valor de referencia.

Para conocer la posición de los cuartiles en el grupo de datos, se debe identificar si la cantidad de datos es par o impar. Para el caso de la vista desarrollador, se cuenta con un conjunto de datos par (28), por lo tanto, se aplica la fórmula presentada en la Figura 32.

$$Q_k = \frac{k * n}{4}$$

Figura 32. Cálculo Posición cuartiles, grupo par de datos

La aplicación de la fórmula expresada en la Figura 32, permitió obtener la posición de cada cuartil: Q1, Q2 y Q3. Esto permitió dividir en 4 partes iguales todo el conjunto de datos no agrupados de la vista desarrollador. En la Figura 32, k corresponde al cuartil que se desea conocer y n al número total de atributos de la vista. La Tabla 40 presenta la posición y valor de los cuartiles para la vista desarrollador.

Tabla 40. Posición de cuartiles vista desarrollador

Cuartil	k	n	Posición	Valor
Q1	1	28	7	92
Q2	2	28	14	108
Q3	3	28	21	116

Para identificar las posiciones de manera correcta, los datos fueron organizados de menor a mayor. En la Figura 33 se presentan los cuartiles obtenidos.

Posición	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	
Atributo		Documentation	Virtualization technique	Response time	Reusability	Jitter	Service use rate	Failure rate	Interoperability	VM Shutdown time	Understandability	Scalability range	Timeliness	Customizability	Latency	Portability	Platforms supported	Changeability	Operating system support	VM Startup time	Service time	API	Operability	Time efficiency	Maintainability	Flexibility	Virtual infrastructure automation	Scalability speed	Down time
Sumatoria	76	76	79	80	83	87	92	94	101	101	103	104	107	108	108	109	112	114	114	115	116	117	118	119	122	123	131	133	
							Q1							Q2							Q3								

Figura 33. Cuartiles vista desarrollador

Para esta vista, los atributos de calidad seleccionados con mayor frecuencia por los desarrolladores son aquellos que están agrupados entre la posición 22 y la posición 28. Los atributos entre la posición 15 y 28 serán llamados en adelante primer y segundo grupo, estos dos grupos cubren el 50% de los atributos que con mayor frecuencia son tenidos en cuenta al realizar el despliegue de una aplicación en la nube.

6.7.1.2. Vista usuario

De los 121 atributos del modelo, 25 corresponden a la vista del usuario. La Tabla 41 presenta el ranking de los resultados obtenidos.

Tabla 41. Resultados vista usuario

Atributo	Sumatoria
<input type="checkbox"/> Continuity	131
<input type="checkbox"/> Self-service	119
<input type="checkbox"/> Interaction	118
<input type="checkbox"/> Waiting time	118
<input type="checkbox"/> Transparency	117
<input type="checkbox"/> Cloud Customer Confidence Level	114
<input type="checkbox"/> Billing process	113
<input type="checkbox"/> Measurability	112
<input type="checkbox"/> VM cost	107
<input type="checkbox"/> Risks	106
<input type="checkbox"/> Data transfer cost	104
<input type="checkbox"/> Client Interface	98
<input type="checkbox"/> Appropriateness Recognizability	98
<input type="checkbox"/> Provider certifications	94
<input type="checkbox"/> Free Trial	94
<input type="checkbox"/> Data geographic /political	91
<input type="checkbox"/> Ease of doing business	90
<input type="checkbox"/> Contracting experience	89

Atributo	Sumatoria
Storage cost	89
Reputation	83
Group Formation	78
Satisfaction	78
Courtesy	73
Reliability Completeness	61
Share	58

Para la vista usuario, el atributo de calidad más relevante fue *Continuity*, y el de menos importancia fue *Share*. En esta vista, la cantidad de datos es impar (25), por lo tanto, se aplica la fórmula presentada en la Figura 34.

$$Q_k = \frac{k * (n + 1)}{4}$$

Figura 34. Cálculo Posición cuartiles, grupo impar de datos

La aplicación de la fórmula expresada en la Figura 34, permitió obtener la posición de cada cuartil: Q1, Q2 y Q3. Nuevamente, en la Figura 34, *k* corresponde al cuartil que se desea conocer y *n* al número total de atributos de la vista.

Con la aplicación de la fórmula presentada en la Figura 34, se obtuvo que la posición para Q1 y Q3 no corresponden de manera absoluta a una posición del listado de atributos, debido a que presenta valores decimales, tal como se presenta en la Tabla 42.

Tabla 42. Posición de cuartiles vista usuario

Cuartil	<i>k</i>	<i>n</i>	Posición
Q1	1	25	6,5
Q2	2	25	13
Q3	3	25	19,5

Por lo anterior, se realizó un proceso de interpolación propuesto en la Figura 35 para conocer el valor exacto de los cuartiles Q1 y Q3.

$$\text{Valor } Q_k = P_1 + (P_2 - P_1) * \text{decimales}$$

Figura 35. Fórmula para realizar la interpolación en el cálculo del valor del cuartil

En la Figura 35 k corresponde al cuartil, P_1 al valor de la posición obtenida sin el residuo decimal, P_2 al valor de la siguiente posición y los decimales de la posición inicial. En consecuencia, para calcular el valor para el cuartil 1 se realiza la operación presentada en la Figura 36.

$$\text{Valor } Q_1 = 83 + (89 - 83) * 0,5 = 86$$

Figura 36. Calculo valor Q1 con interpolación

En la Tabla 43 se presentan los valores de los cuartiles para la vista usuario.

Tabla 43. Valor de cuartiles vista usuario

Cuartil	Valor
Q1	86
Q2	98
Q3	113,5

Ahora bien, la representación de los grupos definidos por cada cuartil para la vista usuario, se presentan en la Figura 37.

Posición	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
Atributo	Share	Completeness	Courtesy	Group Formation	Satisfaction	Reputation	Contracting experience	Storage cost	Ease of doing business	Data geographic /political	Provider certifications	Free Trial	Client Interface	Appropriateness Recognizability	Data transfer cost	Risks	VM cost	Measurability	Billing process	Cloud Customer Confidence Level	Transparency	Interaction	Waiting time	Self-service	Continuity
Sumatoria	58	61	73	78	78	83	89	89	90	91	94	94	98	98	104	106	107	112	113	114	117	118	118	119	131

Q1 = 86
Q2
Q3 = 113,5

Figura 37. Cuartiles vista usuario

Para esta vista, los atributos de calidad seleccionados con mayor frecuencia por los desarrolladores son aquellos que obtuvieron un valor superior a 113,5. Los atributos de la posición 14 en adelante serán llamados primer y segundo grupo. Estos dos grupos cubren el 50% de los atributos que obtuvieron una mayor frecuencia al momento de aplicar los instrumentos.

6.7.1.3. Vista proveedor

De los 121 atributos del modelo, 68 corresponden a la vista del proveedor. La Tabla 44 presenta los resultados obtenidos.

Tabla 44. Resultados vista proveedor

Atributo	Sumatoria
Recoverability	132
Confidentiality	126
Accessibility	125
Data Integrity	124
Successability	124
Resilience	122
Task integrity	120
Up time	119
Stability	117
Professionalism	116
Data Confidentiality	116
Revocation	115
Data consistency	115
Technology change	114
Backup	114
Data transfer	113
Correctness	113
Mean time to quality repair	111
Data Availability	110
Proactive threat and vulnerability management	110
Authentication and Authorization	108
Data accuracy	107
Attribute based Encryption	107
Marginal cost	104
Accuracy	103
Memory read	102
Resource Provisioning efficiency	102
Coverage of failure recovery	102
Initiative	102
Data Sanitization	100
Provider contract / SLA verification	99
Coverage of fault tolerance	97
Utilization	97
CPU speed	96
Safety	96

Atributo	Sumatoria
Memory write	96
Synchronization	95
Punctuality	94
Revision	94
Visibility	91
Adaptability	90
Obligations	88
Identity Management and Authentication	88
Security management	87
Specific service	86
Governance	85
Provider personnel requirements	84
Disk write	84
Outbound	83
Customer Support facility	82
CPU occupancy rate	82
Compliance	81
Functionality	80
Traceability	79
Disk read	79
Version Control	78
Inbound	78
Physical and environmental security	77
Dependability	77
Data correctness	73
Dedicated hardware	63
Liquidated damage	61
Packet loss	60
User error protection	60
Number of SLO violation	58
Sustainability	49
Provider supply chain	48
Network security support	48

Para la vista proveedor, el atributo de calidad usado con mayor frecuencia por los integrantes de la muestra fue *Recoverability*, y el menos empleado fue *Network security support*. El ranking permite utilizar medidas de posición para clasificar la información en cuartiles. Teniendo en cuenta que es un grupo de datos par (68) se aplica la fórmula de la Figura 32.

La Tabla 45 presenta la posición y valor de los cuartiles para la vista proveedor.

Tabla 45. Posición de cuartiles vista desarrollador

Cuartil	<i>k</i>	<i>n</i>	Posición	Valor
Q1	1	68	17	81
Q2	2	68	35	96
Q3	3	68	51	111

Para identificar las posiciones de manera correcta, los datos se organizaron de menor a mayor. En la Figura 38 se presentan los cuartiles.

Posición	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34		
Atributo		Provider supply chain	Network security support	Sustainability	Number of SLO violation	Packet loss	User error protection	Liquidated damage	Dedicated hardware	Data correctness	Physical and environmental security	Dependability	Version Control	Inbound	Traceability	Disk read	Functionality	Compliance	Customer Support facility	CPU occupancy rate	Outbound	Provider personnel requirements	Disk write	Governance	Specific service	Security management	Obligations	Identity Management and Authentication	Adaptability	Visibility	Punctuality	Revision	Synchronization	CPU speed	Safety	
Sumatoria	48	48	49	58	60	60	61	63	73	77	77	78	78	79	79	80	81	82	82	83	84	84	85	86	87	88	88	90	91	94	94	95	96	96	96	
																		Q1																		Q2
Posición	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68		
Atributo	Memory write	Coverage of fault tolerance	Utilization	Provider contract / SLA verification	Data Sanitization	Memory read	Resource Provisioning efficiency	Coverage of failure recovery	Initiative	Accuracy	Marginal cost	Data accuracy	Attribute based Encryption	Authentication and Authorization	Data Availability	Proactive threat and vulnerability management	Mean time to quality repair	Data transfer	Connectness	Technology change	Backup	Revocation	Data consistency	Professionalism	Data Confidentiality	Stability	Up time	Task integrity	Resilience	Data Integrity	Successability	Accessibility	Confidentiality	Recoverability		
Sumatoria	96	97	97	99	100	102	102	102	102	103	104	107	107	108	110	110	111	113	113	114	114	115	115	116	116	117	119	120	122	124	124	125	126	132	132	
																	Q3																			

Figura 38. Cuartiles vista proveedor

Para esta vista, los atributos de calidad seleccionados con mayor frecuencia por los desarrolladores son aquellos que están por encima de la posición 51. Los atributos por encima de la posición 34, el cual corresponde a la posición del cuartil 2, es equivalente a la mediana del grupo de datos. Metodológicamente los grupos con valores superiores a Q2 y Q3 también se denominaron primer y segundo grupo. Estos dos grupos cubren el 50% de los atributos que con mayor frecuencia son tenidos en cuenta al realizar el despliegue de una aplicación en la nube.

6.7.2. Contexto academia

Esta prueba de campo sirvió para identificar los atributos de calidad que reconoce un estudiante universitario del área de informática y sistemas. Los hallazgos de la aplicación del instrumento en las universidades fueron:

- 54%, es decir, 65 de los 121 atributos de calidad definidos en el instrumento eran desconocidos para los estudiantes.
- Algunos estudiantes manifestaron no tener conocimiento sobre la gran mayoría de los atributos de calidad presentados, sin embargo, trataron de responder el instrumento con base en el nombre y la pregunta.
- Al finalizar el instrumento, los estudiantes fueron abordados para contestar la siguiente pregunta: ¿Ha realizado el despliegue de una aplicación en algún proveedor de servicios en la nube? El 76% de los participantes respondió negativamente, e indicaron que el conocimiento que tienen sobre el tema es netamente teórico.

La aplicación del instrumento en el sector académico no fue tomada en cuenta para la construcción del ranking de los atributos de calidad. No obstante, a la luz de los resultados, es factible sugerir un proceso de revisión curricular en los programas de ingeniería de sistemas y afines, que permita fortalecer la temática de computación en la nube en todo su contexto.

6.8. Guía para la selección de atributos de calidad

Con los resultados de la validación se identificaron los atributos de calidad que con frecuencia utilizan los desarrolladores al construir aplicaciones que van a ser desplegadas en la nube, de esta manera se estableció un orden definido por cuartiles. Con base en el modelo propuesto y el proceso de validación, se desarrolló una guía que permitirá a los desarrolladores seleccionar y hacer seguimiento al cumplimiento de los atributos de calidad por parte de un proveedor.

Al igual que el modelo propuesto, la guía consta de 3 vistas: desarrollador, proveedor y usuario. Cada vista contiene la totalidad de los atributos de calidad definidos por el modelo. Para la vista desarrollador existen 28 atributos, para la vista usuario 25 atributos y para la vista proveedor 68 atributos, para un total de 121 atributos de calidad.

En cada vista, los atributos están organizados de acuerdo con el ranking obtenido en la tabulación de los instrumentos. Para cada uno se presenta la información de la posición en el ranking y descriptores definidos en la Tabla 46.

Tabla 46. Descriptores de la guía para la selección de atributos de calidad

Descriptor	Significado
Nro.	Posición en el ranking de atributos para la vista
Atributo	Presenta el nombre del atributo de calidad

Descriptor	Significado
Pregunta	Contiene la pregunta registrada en el instrumento de validación aplicado a los desarrolladores
Peso	Representa la importancia que tiene el atributo para el proyecto o desarrollo en el cual será utilizado. La escala para el peso está entre 0 y 5, 0 indica que no se tendrá en cuenta el atributo y 5 que es de vital importancia
Cumple	Esta calificación se asigna de manera posterior al uso del servicio, representa el cumplimiento por parte del proveedor en lo relacionado al atributo seleccionado, los valores disponibles son sí y no
Comentarios	Espacio donde se puede describir algo relevante del atributo

6.8.1. Vista desarrollador

Como se definió en el alcance del proyecto de investigación, la presente guía, busca ser un instrumento útil en dos fases: en la fase de requerimientos, para apoyar la construcción de la especificación de requisitos y en la fase de pruebas, para verificar el nivel de cumplimiento por parte del proveedor. La Figura 39 presenta un fragmento de la guía para la vista desarrollador. Este fragmento evidencia los atributos organizados en grupos, tal como fueron clasificados por los cuartiles obtenidos en la tabulación de los datos.

La guía sugiere el uso de los atributos de acuerdo con el ranking establecido en el modelo, sin embargo, será el desarrollador quien decida finalmente los atributos a utilizar, esto se debe a que cada proyecto software tiene sus propias características.

TQACA Extended Quality Model Modelo de Calidad Extendido para la Taxonomía de Atributos de Calidad en Aplicaciones para la Nube					
Nro.	Atributo	Pregunta	Peso	Cumple	Comentarios
28	Down time	¿Considera importante que ante una caída del servicio de la nube su aplicación vuelva funcionar en el menor tiempo posible?	3	Si	Grupo 1: 25 % de los atributos con valor superior al Q3 obtenido.
27	Scalability speed	¿Le interesa que las solicitudes de escalabilidad manuales o automáticas se realicen en el menor tiempo posible?	5	Si	
22	Operability	¿Tiene en cuenta que los recursos ofrecidos por el proveedor puedan ser utilizados por muchos usuarios de manera simultánea?	0	No	
21	API	¿Es importante para usted que el proveedor proporcione APIs con documentación adecuada para el uso de los servicios?	0	No	Grupo 2: Atributos con valor superior al Q2 y hasta el Q3.
18	VM Startup time	¿Usted considera importante que los tiempos de respuesta de la máquina virtual al encenderse sean mínimos?	0	No	
15	Portability	¿Tiene en cuenta la capacidad del proveedor para permitirle migrar su aplicación desplegada a otra máquina con diferente sistema operativo?	0	No	Grupos 3 y 4: Atributos con valor hasta Q2
14	Latency	¿Usted analiza los tiempos de respuesta del servidor o la máquina virtual ante el uso de su aplicación?	0	No	
7	Failure rate	¿Verifica los reportes suministrados por su proveedor acerca de las fallas que ocurren en lapsos?	0	No	
3	Response time	¿Tiene en cuenta los tiempos de respuesta en milisegundos que puede llegar a tardar la máquina virtual al dar una respuesta de su aplicación?	0	No	
1	Documentation	¿Cuándo analiza la documentación de los servicios, tiene en cuenta la forma como está publicada la documentación?	0	No	
			10	Porcentaje de cumplimiento = 80%	

Figura 39. Guía vista desarrollador

En la última fila de la guía se presenta el porcentaje de cumplimiento de los atributos de calidad.

6.8.2. Vista usuario

La Figura 40 presenta un fragmento de la guía para la vista usuario. Los atributos están clasificados por grupos, tal como fueron ubicados por los cuartiles.

TQACA Extended Quality Model Modelo de Calidad Extendido para la Taxonomía de Atributos de Calidad en Aplicaciones para la Nube					
Nro.	Atributo	Pregunta	Peso	Cumple	Comentarios
25	Continuity	¿Considera relevante la continuidad de los recursos suministrados por el proveedor?	3	Si	Grupo 1: 25% de los atributos con valor superior al Q3 obtenido.
24	Self-service	¿Tiene en cuenta que pueda autogestionar todos los servicios que requiera la aplicación a desplegar?	2	Si	
20	Cloud Customer Confidence Level	¿Usted analiza el nivel de confianza del proveedor antes de realizar un despliegue?	0	No	
19	Billing process	¿Es importante para usted que el proveedor le presente facturas detalladas del uso de sus recursos?	2	Si	Grupo 2: Atributos con valor superior al Q2 y hasta el Q3.
15	Data transfer cost	¿Para usted es importante el costo de la transferencia de los datos?	0	No	
14	Appropriateness Recognizability	¿Usted tiene en cuenta la documentación, y listado de características proporcionadas por el proveedor, para identificar si lo ofrecido realmente se adapta a sus necesidades?	0	No	
13	Client interface	¿Tiene en cuenta las interfaces gráficas suministradas por el proveedor para realizar el despliegue de sus aplicaciones?	2	No	Grupos 3 y 4: Atributos con valor hasta Q2
8	Storage cost	¿Considera los costos del almacenamiento secundario?	0	No	
7	Contracting experience	¿Usted tiene en cuenta experiencias anteriores para cambiar el proveedor en la nube?	0	No	
2	Completeness	¿Usted verifica que el proveedor haga seguimiento a los servicios y máquinas virtuales utilizadas por su aplicación?	3	Si	
1	Share	¿A usted le interesa que los servicios que utilice sean compartidos con personas externas en la misma instancia que usted los utiliza?	0	No	
			12	Porcentaje de cumplimiento = 83%	

Figura 40. Guía vista usuario

6.8.3. Vista proveedor

La Figura 41 presenta un fragmento de la guía para la vista proveedor. Los atributos están clasificados por grupos, tal como fueron ubicados por los cuartiles.

TQACA Extended Quality Model Modelo de Calidad Extendido para la Taxonomía de Atributos de Calidad en Aplicaciones para la Nube					
Nro.	Atributo	Pregunta	Peso	Cumple	Comentarios
68	Recoverability	¿Considera importante que, ante un desastre del proveedor, su aplicación esté funcionando en el menor tiempo posible?	5	Si	Grupo 1: 25% de los atributos con valor superior al Q3 obtenido.
67	Confidentiality	¿Le preocupa que los datos gestionados por su aplicación estén disponibles solo para los usuarios autorizados por usted?	0	No	
54	Technology change	¿Usted tiene en cuenta que el proveedor haga cambios tecnológicos que puedan llegar a afectar su aplicación?	0	No	
52	Data transfer	¿La velocidad de transferencia de los datos es importante para usted?	0	No	Grupo 2: Atributos con valor superior al Q2 y hasta el Q3.
51	Mean time to quality repair	Ante sobrecargas por el uso de la aplicación desplegada, ¿Considera importante que su proveedor suministre una carga adicional de trabajo de manera automática?	2	No	
47	Attribute based Encryption	¿Es importante que los datos transferidos desde y hacia su aplicación estén cifrados por defecto por el proveedor?	1	No	
35	Memory write	¿Tiene en cuenta la velocidad de escritura de datos en la memoria?	0	No	Grupos 3 y 4: Atributos con valor hasta Q2
34	Safety	¿Es importante para usted que se utilicen sistemas de seguridad tipo token, Marca de tiempo, Firmas o cifrado para el acceso a sus recursos?	0	No	
17	Compliance	¿Usted se asegura que el proveedor de la nube cumpla cabalmente con los estándares definidos?	0	No	
1	Provider supply chain	¿Usted tiene en cuenta que el proveedor tenga contratos con terceros para ofrecerle servicios?	0	No	
			8	Porcentaje de cumplimiento = 62%	

Figura 41. Guía vista proveedor

7. CONCLUSIONES

7.1. Conclusiones por objetivo

La Tabla 47 describe los objetivos planteados en la presente tesis, el nivel de cumplimiento y las conclusiones asociadas a cada objetivo.

Tabla 47. Relación objetivos, cumplimiento y conclusiones

Nro.	Objetivo	Cumplimiento	Conclusiones
1	Comparar los estándares y mejores prácticas para la construcción de vocabularios controlados y taxonomías, que permita determinar el esquema de etiquetado, clasificación e indexación.	Se analizaron los trabajos de (Hedden, 2010), (Mell & Grance, 2011; NIST, 2011a, 2011b) y (CSMIC, 2014b), para establecer los conceptos, principios y estructura de la taxonomía. Se dio cumplimiento a la normatividad en cuanto visualización y relaciones y se estableció el alcance y la forma de acuerdo con los estándares para identificar el esquema de etiquetado e indexación. El resultado de este objetivo se ve reflejado en los apartados: 2 TRABAJOS RELACIONADOS, 3 VOCABULARIOS CONTROLADOS y 5 PRESENTACIÓN DE LA TAXONOMÍA.	Para la construcción de taxonomías es de obligatorio cumplimiento establecer una línea de base, que permita organizar el conocimiento sin desconocer el trabajo realizado en un área en particular.
2	Diseñar un protocolo para crear un vocabulario controlado sobre atributos de calidad en aplicaciones diseñadas para ser desplegadas en la nube, de acuerdo con las mejores prácticas y estándares definidos en la literatura científica para la construcción de vocabularios.	El protocolo fue creado de acuerdo con los lineamientos propuestos por Kitchenham, el cual permitió realizar un seguimiento detallado a todo el proceso de construcción de la taxonomía, aportó las evidencias necesarias para dar credibilidad y transparencia a todo el trabajo y permitió repetir paso a paso el trabajo realizado por parte de un investigador externo. El resultado de este objetivo se presenta en apartado 4 PROTOCOLO PARA LA CONSTRUCCIÓN DEL VOCABULARIO.	La construcción del protocolo se convirtió en una herramienta de control y seguimiento para el desarrollo del proyecto. Adicionalmente, sirvió como herramienta de auditoría al trabajo realizado.

Nro.	Objetivo	Cumplimiento	Conclusiones
3	Evaluar estudios sobre la creación y uso de vocabularios controlados, taxonomías, tesauros y ontologías en computación en la nube, para identificar la articulación del aporte del presente estudio en la organización del conocimiento.	<p>Se realizó la evaluación de los trabajos relacionados con vocabularios controlados, taxonomías y ontologías, ampliando el espectro a trabajos diferentes a computación en la nube. Adicionalmente, se incluyeron modelos de calidad, puesto que tenían como base fundamental la agrupación de atributos de calidad.</p> <p>El apartado 2 TRABAJOS RELACIONADOS, presenta el análisis realizado sobre la temática de estudio.</p>	El aporte de esta tesis en la organización del conocimiento es un 42% de características de calidad adicionales con respecto al trabajo desarrollado por la Universidad Carnegie Mellon (CSMIC, 2014b).
4	Construir un vocabulario controlado sobre atributos de calidad, que permita organizar el conocimiento recopilado en la temática y soporte el desarrollo de la taxonomía, utilizando el protocolo diseñado en el segundo objetivo específico de la presente propuesta.	<p>Se realizó la construcción del vocabulario controlado y se construyó un instrumento de obligatorio cumplimiento para la publicación de la taxonomía. La taxonomía está soportada en la garantía literaria que se definió en el protocolo.</p> <p>El apartado 5 PRESENTACIÓN DE LA TAXONOMÍA y los Apéndice A y Apéndice B contienen la información detallada.</p>	El conocimiento en el área de atributos de calidad para computación en la nube recibe un gran aporte por parte del desarrollo de esta investigación. Si bien el resultado se orienta a los atributos al despliegue de aplicaciones, sin embargo, pueden ser utilizados por otros investigadores en contextos diferentes como la selección de proveedores o el ranking de servicios.
5	Construir a partir de la taxonomía creada, un modelo de calidad compuesto por métricas, que pueda ser utilizado en la fase de especificación de requerimientos de un proceso de desarrollo software.	<p>El modelo de calidad propuesto es una extensión de la taxonomía, por lo tanto, contiene una estructura jerárquica con definición de métricas. Adicionalmente, generó una guía para la selección de atributos de calidad en la fase de requerimientos.</p> <p>El modelo incluye instrumentos y herramientas de visualización por vistas.</p> <p>El apartado 6 presenta el modelo de calidad.</p>	La articulación de las vistas en el modelo de calidad le permite a un desarrollador identificar los atributos de calidad más utilizados a la hora de realizar el despliegue de una aplicación en la nube.
6	Validar la taxonomía propuesta mediante la	La validación de la taxonomía se realizó de conformidad con la garantía literaria y la verificación de la	El muestreo no probabilístico utilizado para la validación del modelo permitió

Nro.	Objetivo	Cumplimiento	Conclusiones
	utilización del modelo de calidad construido en el objetivo anterior, en dos contextos: industria software y academia. Para analizar el impacto de la taxonomía en proyectos de desarrollo, bajo la supervisión de un panel de expertos.	organización por parte de expertos en desarrollo. Como evidencia se encuentra el protocolo desarrollado en el objetivo 2. El modelo también permitió realizar la validación de la taxonomía en dos contextos: industria y academia.	focalizar el trabajo en tres grupos de desarrolladores. La identificación y reconocimiento por parte de los desarrolladores de los atributos de calidad fue total. En el contexto academia, por el contrario, se evidenciaron oportunidades de mejora en los procesos de formación de los futuros profesionales.

7.2. Conclusiones respecto a la hipótesis

Con relación a las preguntas de investigación y las hipótesis planteadas en la propuesta de investigación, se presentan las conclusiones en la Tabla 48.

Tabla 48. Conclusiones respecto a las hipótesis

Pregunta de investigación	Hipótesis	Conclusión
¿Un vocabulario controlado sobre atributos de calidad en computación en la nube podría contribuir a la organización del conocimiento en este ámbito?	Un vocabulario controlado sobre atributos de calidad en computación en la nube contribuye a la organización del conocimiento en este ámbito.	La hipótesis descriptiva es válida. TQACA permitió organizar el conocimiento académico para presentarlo a un grupo focalizado de la industria, obteniendo como resultado la aceptación de la totalidad de los atributos definidos en una estructura jerárquica.
¿Es posible diseñar un protocolo, que permita construir una clasificación del conocimiento sobre atributos de calidad en aplicaciones diseñadas para ser desplegadas en la nube?	El diseño de un protocolo permite construir una clasificación del conocimiento sobre atributos de calidad en aplicaciones diseñadas para ser desplegadas en la nube.	Esta hipótesis de causalidad es válida. El protocolo sirvió como hoja de ruta para que tanto el investigador principal como sus colaboradores tuvieran a la mano un instrumento que permitiera realizar el seguimiento a todo el proceso de construcción de la taxonomía.
¿Una taxonomía permitiría la definición de conceptos y relaciones para establecer un modelo unificado, que permita el análisis de atributos de	La definición de una taxonomía, soportada en estructura jerárquica de conceptos y sus relaciones permite establecer un modelo para	La hipótesis es válida. El modelo generado en esta tesis en una extensión de la taxonomía definida. El análisis de atributos realizado por los desarrolladores permitió corroborar la importancia de los atributos de calidad al

Pregunta de investigación	Hipótesis	Conclusión
calidad en aplicaciones diseñadas para ser desplegadas en la nube?	el análisis de atributos de calidad en aplicaciones diseñadas para ser desplegadas en la nube.	momento de definir requerimientos no funcionales en el desarrollo de software para la nube.

7.3. Conclusiones generales

- Para construir un cuerpo de conocimiento se debe seleccionar una línea de base que orientar la investigación. En el presente estudio las líneas fueron los trabajos de: Heather Hedden en “The Accidental Taxonomist”, la guía Z39.19-2005 publicada por ANSI/NISO (ANSI/NISO, 2005) y Service Measurement Index (SMI).
- No se encontró evidencia de un cuerpo de conocimiento unificado sobre la temática de atributos de calidad en aplicaciones a ser desplegadas en la nube, por tal motivo, la presente tesis ofrece una base de conocimiento, que propone una taxonomía con términos y atributos de calidad desde tres perspectivas: proveedor, cliente y desarrollador.
- La presentación y búsqueda de información en la taxonomía construida, requiere de un instrumento que permita compartir el resultado de la investigación, por tal motivo, se desarrolló una plataforma web que permitió la captura, visualización, edición y búsqueda de atributos de calidad.
- A partir del proceso de revisión sistemático de literatura, se encontró un grupo de modelos para la evaluación de los servicios en la nube, sin embargo, cada uno de ellos propone sus propios esquemas de evaluación sin llegar a una unificación de criterios. El presente estudio entrega a la comunidad científica un modelo creado a partir de una taxonomía de atributos de calidad que unifica los atributos requeridos por los actores involucrados en el despliegue de aplicaciones en la nube.
- Como resultado del proceso de validación del modelo, se recopiló información pertinente sobre los tópicos relevantes que pueden ser incluidos en la formación de los futuros profesionales del área de sistemas y afines.

- El grupo de investigación GIDATIC, tiene entre sus líneas de investigación la Ingeniería del Software. Los resultados de esta tesis están articulados con esta línea de investigación y se pueden considerar como un aporte en la organización del conocimiento del área de calidad en computación en la nube.
- Entre los objetivos del grupo de investigación GIDATIC está el desarrollo e implementación de aplicaciones que permitan analizar y extraer información, mediante diferentes metodologías. El presente estudio aporta al grupo de investigación, herramientas para la organización, manipulación y el despliegue jerárquico de los atributos de calidad en computación en la nube.

8. TRABAJOS FUTUROS

A partir de la investigación realizada y de las conclusiones presentadas en el apartado anterior, se plantean los posibles trabajos a futuro.

- En el proceso de extracción de datos se evidenció que la mayor parte de los trabajos realizados por la comunidad científica estaban orientados a los atributos: disponibilidad (*availability*) y tiempo de respuesta (*response time*). Esta particularidad evidencia la importancia que tiene la prestación del servicio por parte de los proveedores y genera un espacio de estudio para proponer acuerdos de nivel de servicio que permitan a los usuarios y desarrolladores tener un control sobre este tipo de atributos.
- La guía para la selección de atributos de calidad en la fase de requerimientos requiere de un instrumento que automatice su proceso y permita llevar un seguimiento al despliegue de las aplicaciones por parte de los desarrolladores.
- La definición de métricas nuevas para los atributos identificados en la taxonomía se abre como un problema abierto, toda vez que muchos de los atributos aún no pueden ser calculados en contextos de proveedores privados y/o públicos.
- Existe una oportunidad de mejora evidente en la academia en la apropiación y transferencia de conocimiento a los estudiantes. La articulación entre teoría y práctica debe estar cohesionada para que los estudiantes reconozcan los conceptos en los dos contextos.

9. REFERENCIAS

- Abdeladim, A., Baina, S., & Baina, K. (2014). Elasticity and scalability centric quality model for the cloud, 135–140.
- Abderrahim, W., & Choukair, Z. (2018). Brokerage-based dependability integration in cloud computing services. *Journal of Supercomputing*, 74(7), 3359–3387. <https://doi.org/10.1007/s11227-018-2388-4>
- Abuhussein, A., Alsubaei, F., Shiva, S., & Sheldon, F. T. (2016). Evaluating Security and Privacy in Cloud Services. *Proceedings - International Computer Software and Applications Conference*, 1(1), 683–686. <https://doi.org/10.1109/COMPSAC.2016.263>
- Ahmed, M., Liu, L., Hardy, J., & Yuan, B. (2014). An efficient algorithm for partially matched web services based on consumer's QoS requirements. *Proceedings - 2014 IEEE/ACM 7th International Conference on Utility and Cloud Computing, UCC 2014*, 859–864. <https://doi.org/10.1109/UCC.2014.140>
- Ahmed, W., & Wu, Y. W. (2013). A survey on reliability in distributed systems. *Journal of Computer and System Sciences*, 79(8), 1243–1255. <https://doi.org/10.1016/j.jcss.2013.02.006>
- Alabool, H., Kamil, A., Arshad, N., & Alarabiat, D. (2018). Cloud service evaluation method-based Multi-Criteria Decision-Making: A systematic literature review. *Journal of Systems and Software*, 139, 161–188. <https://doi.org/10.1016/j.jss.2018.01.038>
- Aljazzaf, Z. (2015). Bootstrapping quality of web services. *Journal of King Saud University - Computer and Information Sciences*, 27(3), 323–333. <https://doi.org/10.1016/j.jksuci.2014.12.003>
- Alkhanak, E. N., Lee, S. P., & Khan, S. U. R. (2015). Cost-aware challenges for workflow scheduling approaches in cloud computing environments: Taxonomy and opportunities. *Future Generation Computer Systems*, 50, 3–21. <https://doi.org/10.1016/j.future.2015.01.007>
- Almutiry, O., Wills, G., Alwabel, A., Crowder, R., & Walters, R. (1998). Toward A Framework For Data Quality In Cloud? Based Health Information System. *J. Chem. Soc. Faraday T.*, 94(12), 1687–1693.
- Alnanih, R., Ormandjieva, O., & Radhakrishnan, T. (2013). A New Quality-in-Use Model for Mobile User Interfaces. En *2013 Joint Conference of the 23rd International Workshop on Software Measurement and the 8th International Conference on Software Process and Product Measurement* (pp. 165–170). IEEE. <https://doi.org/10.1109/IWSM-Mensura.2013.32>
- Aloraini, A., & Hammoudeh, M. (2017). A Survey on Data Confidentiality and Privacy in Cloud Computing. *Proceedings of the International Conference on Future Networks and Distributed Systems - ICFNDS '17*, 1–7. <https://doi.org/10.1145/3102304.3102314>

- ANSI/NISO. (2005). *Guidelines for the Construction , Format , and Management of Monolingual Controlled Vocabularies. ANSI/NISO Z39.19-2005 (R2010)* (Vol. 2005).
- Ardagna, C. A., Damiani, E., Frati, F., Rebecani, D., & Ughetti, M. (2012). Scalability Patterns for Platform-as-a-Service. En *2012 IEEE Fifth International Conference on Cloud Computing* (pp. 718–725). IEEE. <https://doi.org/10.1109/CLOUD.2012.41>
- Aruna, L., & Aramudhan, M. (2016). Framework for Ranking Service Providers of Federated Cloud Architecture using Fuzzy Sets. *International Journal of Technology*, 7(4), 643. <https://doi.org/10.14716/ijtech.v7i4.1498>
- Assis, M. R. M., & Bittencourt, L. F. (2016). A survey on cloud federation architectures: Identifying functional and non-functional properties. *Journal of Network and Computer Applications*, 72, 51–71. <https://doi.org/10.1016/j.jnca.2016.06.014>
- Baliyan, N., & Kumar, S. (2013). Quality assessment of software as a service on cloud using fuzzy logic. *2013 IEEE International Conference on Cloud Computing in Emerging Markets, CCEM 2013*. <https://doi.org/10.1109/CCEM.2013.6684439>
- Bao, D., Xiao, Z., Sun, Y., & Zhao, J. (2010). A method and framework for quality of cloud services measurement. En *2010 3rd International Conference on Advanced Computer Theory and Engineering(ICACTE)* (pp. V5-358-V5-362). IEEE. <https://doi.org/10.1109/ICACTE.2010.5579535>
- Baranwal, G., & Vidyarthi, D. P. (2014). A framework for selection of best cloud service provider using ranked voting method. *Souvenir of the 2014 IEEE International Advance Computing Conference, IACC 2014*, 831–837. <https://doi.org/10.1109/IAdCC.2014.6779430>
- Baranwal, G., & Vidyarthi, D. P. (2016). A cloud service selection model using improved ranked voting method. *Concurrency and Computation: Practice and Experience*, 28(13), 3540–3567. <https://doi.org/10.1002/cpe.3740>
- Barité-Roqueta, M. (2011). Sistemas de organización del conocimiento: Una tipología actualizada. *Informação & Informação*, 16(3), 122–139. <https://doi.org/10.5433/1981-8920.2011v16nesp.p122>
- Barité, M. (2009). Garantía literaria y normas para construcción de vocabularios controlados : aspectos epistemológicos y metodológicos. *Scire*, 15(2), 13–24.
- Bassil, Y. (2012). No Title, 3(1), 1–6. Software Engineering. Recuperado de <http://arxiv.org/abs/1204.0195>
- Becker, M., Lehrig, S., & Becker, S. (2015). Systematically Deriving Quality Metrics for Cloud Computing Systems. *Proceedings of the 6th ACM/SPEC International Conference on Performance Engineering - ICPE '15*, 169–174. <https://doi.org/10.1145/2668930.2688043>
- Belalem, G., Bouamama, S., & Sekhri, L. (2011). An Effective Economic Management of

- Resources in Cloud Computing. *Journal of Computers*, 6(3), 404–411. <https://doi.org/10.4304/jcp.6.3.404-411>
- Ben-Abdallah, E., Boukadi, K., & Hammami, M. (2018). SMI-Based Opinion Analysis of Cloud Services from Online Reviews (pp. 683–692). https://doi.org/10.1007/978-3-319-76348-4_66
- Bilecki, L. F., Fiorese, A., & Matos, F. (2017). A trust reputation architecture for virtual organization integration in cloud computing environment. *ICEIS 2017 - Proceedings of the 19th International Conference on Enterprise Information Systems*, 2(Iceis), 695–702. <https://doi.org/10.5220/0006276506950702>
- Boniface, M., Nasser, B., Papay, J., Phillips, S. C., Servin, A., Yang, X., ... Kyriazis, D. (2010). Platform-as-a-Service Architecture for Real-Time Quality of Service Management in Clouds. En *2010 Fifth International Conference on Internet and Web Applications and Services* (pp. 155–160). IEEE. <https://doi.org/10.1109/ICIW.2010.91>
- Brataas, G., Herbst, N., Ivansek, S., & Polutnik, J. (2017). Scalability Analysis of Cloud Software Services. *Proceedings - 2017 IEEE International Conference on Autonomic Computing, ICAC 2017*, 285–292. <https://doi.org/10.1109/ICAC.2017.34>
- Brereton, P., Kitchenham, B. A., Budgen, D., Turner, M., & Khalil, M. (2007). Lessons from applying the systematic literature review process within the software engineering domain. *Journal of Systems and Software*, 80(4), 571–583. <https://doi.org/10.1016/j.jss.2006.07.009>
- Bruneo, D. (2014). A Stochastic Model to Investigate Data Center Performance and QoS in IaaS Cloud Computing Systems. *IEEE Transactions on Parallel and Distributed Systems*, 25(3), 560–569. <https://doi.org/10.1109/TPDS.2013.67>
- Cai, B. L., Zhang, R. Q., Zhou, X. B., Zhao, L. P., & Li, K. Q. (2017). Experience Availability: Tail-Latency Oriented Availability in Software-Defined Cloud Computing. *Journal of Computer Science and Technology*, 32(2), 250–257. <https://doi.org/10.1007/s11390-017-1719-x>
- Callejas-Cuervo, M., Alarcón-Aldana, A. C., & Álvarez-Carreño, A. M. (2017). Modelos de calidad del software, un estado del arte. *Entramado*, 13(1), 236–250. <https://doi.org/10.18041/entramado.2017v13n1.25125>
- Cao, W., Gao, Y., Lin, B., Feng, X., Xie, Y., Lou, X., & Wang, P. (2018). TcpRT: Instrument and diagnostic analysis system for service quality of cloud databases at massive scale in real-time. *Proceedings of the ACM SIGMOD International Conference on Management of Data*, 615–627. <https://doi.org/10.1145/3183713.3190659>
- Chang, V., Ramachandran, M., Yao, Y., Kuo, Y. H., & Li, C. S. (2016). A resiliency framework for an enterprise cloud. *International Journal of Information Management*, 36(1), 155–166. <https://doi.org/10.1016/j.ijinfomgt.2015.09.008>
- Chatzipetrou, P., Angelis, L., Barney, S., & Wohlin, C. (2011). Software Product Quality in

- Global Software Development: Finding Groups with Aligned Goals. En *2011 37th EUROMICRO Conference on Software Engineering and Advanced Applications* (pp. 435–442). IEEE. <https://doi.org/10.1109/SEAA.2011.67>
- Chauhan, M. A., & Babar, M. A. (2012). Cloud infrastructure for providing tools as a service. En *Proceedings of the WICSA/ECSA 2012 Companion Volume on - WICSA/ECSA '12* (p. 5). New York, New York, USA: ACM Press. <https://doi.org/10.1145/2361999.2362002>
- Chen, K.-T., Chang, Y.-C., Tseng, P.-H., Huang, C.-Y., & Lei, C.-L. (2011). Measuring the latency of cloud gaming systems. *Proceedings of the 19th ACM international conference on Multimedia - MM '11*, 1269. <https://doi.org/10.1145/2072298.2071991>
- Choi, C. R., & Jeong, H. Y. (2014). Quality evaluation and best service choice for cloud computing based on user preference and weights of attributes using the analytic network process. *Electronic Commerce Research*, 14(3), 245–270. <https://doi.org/10.1007/s10660-014-9156-1>
- Colomo-Palacios, R., & Soto-Acosta, P. (2011). Software quality management improvement through mentoring: an exploratory study from GSD projects. *On the Move to Meaningful Internet Systems: OTM 2011 Workshops*, 190–199. https://doi.org/10.1007/978-3-642-25126-9_29
- Colomo-Palacios, Ricardo, & Rodríguez, J. M. Á. (2014). Semantic Representation and Computation of Cloud-Based Customer Relationship Management Solutions (pp. 347–357). https://doi.org/10.1007/978-3-662-45550-0_35
- Comerio, M., Truong, H.-L., Batini, C., & Dustdar, S. (2010). Service-oriented data quality engineering and data publishing in the cloud. En *2010 IEEE International Conference on Service-Oriented Computing and Applications (SOCA)* (pp. 1–6). IEEE. <https://doi.org/10.1109/SOCA.2010.5707184>
- Costa, P., Santos, J. P., & Da Silva, M. M. (2013). Evaluation Criteria for Cloud Services. En *2013 IEEE Sixth International Conference on Cloud Computing* (pp. 598–605). IEEE. <https://doi.org/10.1109/CLOUD.2013.70>
- CSMIC. (2014a). Selecting a cloud provider defining widely accepted measures for cloud services. Recuperado de <https://spark.adobe.com/page/PN39b/>
- CSMIC. (2014b). *Service Measurement Index Framework Version 2.1*. California. Recuperado de http://csmic.org/downloads/SMI_Overview_TwoPointOne.pdf
- Da Cunha Rodrigues, G., Calheiros, R. N., De Carvalho, M. B., Paula Dos Santos, C. R., Granville, L. Z., Tarouco, L., & Buyya, R. (2016). The interplay between timeliness and scalability in cloud monitoring systems. *Proceedings - IEEE Symposium on Computers and Communications, 2016-Febru*, 776–781. <https://doi.org/10.1109/ISCC.2015.7405608>
- De Benedetti, M., Durso, F., Messina, F., Pappalardo, G., & Santoro, C. (2015). A Hybrid

- Model for Ranking Cloud Services. *Proceedings - 2015 10th International Conference on P2P, Parallel, Grid, Cloud and Internet Computing, 3PGCIC 2015*, 504–509. <https://doi.org/10.1109/3PGCIC.2015.28>
- Deshpande, P., Sharma, S. C., Peddoju, S. K., & Abraham, A. (2018). Security and service assurance issues in Cloud environment. *International Journal of Systems Assurance Engineering and Management*, 9(1), 194–207. <https://doi.org/10.1007/s13198-016-0525-0>
- Drucker, A. M., Fleming, P., & Chan, A.-W. (2016). Research Techniques Made Simple: Assessing Risk of Bias in Systematic Reviews. *Journal of Investigative Dermatology*, 136(11), e109–e114. <https://doi.org/10.1016/j.jid.2016.08.021>
- Du, J., & Li, X. (2013). Adaptive and attribute-based trust model for service-level agreement guarantee in cloud computing. *IET Information Security*, 7(1), 39–50. <https://doi.org/10.1049/iet-ifs.2012.0232>
- Dustdar, S., Guo, Y., Satzger, B., & Truong, H.-L. (2011). Principles of Elastic Processes. *IEEE Internet Computing*, 66–71. <https://doi.org/10.1145/1754288.1754304>
- Elizabeth, B. L., Ramya, K., Prakash, A. J., & Uthariaraj, V. R. (2014). Trustworthy mechanisms for selecting cloud service providers. *2014 International Conference on Recent Trends in Information Technology, ICRTIT 2014*. <https://doi.org/10.1109/ICRTIT.2014.6996182>
- Faragardi, H. R., Shojaee, R., Tabani, H., & Rajabi, A. (2013). An analytical model to evaluate reliability of cloud computing systems in the presence of QoS requirements. *2013 IEEE/ACIS 12th International Conference on Computer and Information Science, ICIS 2013 - Proceedings*, 315–321. <https://doi.org/10.1109/ICIS.2013.6607860>
- Ferreira, J. J., Moreno, M. F., Brandao, R., & Cerqueira, R. (2016). Multimedia in Cognitive-Intensive Practices: A Case with ATLAS.ti Supporting HCI Qualitative Research. En *2016 IEEE International Symposium on Multimedia (ISM)* (pp. 679–684). IEEE. <https://doi.org/10.1109/ISM.2016.0144>
- Franke, D., Kowalewski, S., & Weise, C. (2012). A Mobile Software Quality Model. En *2012 12th International Conference on Quality Software* (pp. 154–157). IEEE. <https://doi.org/10.1109/QSIC.2012.49>
- Fraunhofer ISST. (2009). *Guidelines and Good Practices for Taxonomies*. Recuperado de <https://joinup.ec.europa.eu/sites/default/files/document/2011-12/guidelines-and-good-practices-for-taxonomies-v1.3a.pdf>
- Freitas, A. L., Parlavantzas, N., & Pazat, J. L. (2012). An integrated approach for specifying and enforcing SLAs for cloud services. *Proceedings - 2012 IEEE 5th International Conference on Cloud Computing, CLOUD 2012*, 376–383. <https://doi.org/10.1109/CLOUD.2012.135>

- Garg, S. K., Versteeg, S., & Buyya, R. (2013). A framework for ranking of cloud computing services. *Future Generation Computer Systems*, 29(4), 1012–1023. <https://doi.org/10.1016/j.future.2012.06.006>
- Garshol, L. M. (2004). Metadata? Thesauri? Taxonomies? Topic Maps! Recuperado el 6 de junio de 2017, de <http://www.ontopia.net/topicmaps/materials/tm-vs-thesauri.html#N429>
- Ghafori, V., & Manouchehri Sarhadi, R. (2013). Best Cloud Provider Selection using Integrated ANP-DEMATEL and Prioritizing SMI Attributes. *International Journal of Computer Applications*, 71(16), 18–25. <https://doi.org/10.5120/12441-9133>
- Ghahramani, M. H., Zhou, M., & Hon, C. T. (2017). Toward cloud computing QoS architecture: Analysis of cloud systems and cloud services. *IEEE/CAA Journal of Automatica Sinica*, 4(1), 6–18. <https://doi.org/10.1109/JAS.2017.7510313>
- Gholami, A., & Arani, M. G. (2015). A trust model for resource selection in cloud computing environment. En *2015 2nd International Conference on Knowledge-Based Engineering and Innovation (KBEI)* (pp. 144–151). IEEE. <https://doi.org/10.1109/KBEI.2015.7436036>
- Ghosh, N., & Ghosh, S. K. (2012). An approach to identify and monitor SLA parameters for storage-as-a-service cloud delivery model. *2012 IEEE Globecom Workshops, GC Wkshps 2012*, (vi), 724–729. <https://doi.org/10.1109/GLOCOMW.2012.6477664>
- Gonzalez, N. M., Miers, C. C., Red, F. F., Simpl, M., Prof, A., Gualberto, L., & C, T. N. (2011). A taxonomy model for cloud computing services. *Proceedings of the 1st International Conference on Cloud Computing and Services Science*, 56–65. <https://doi.org/10.5220/0003384800560065>
- Grandhi, S., & Wibowo, S. (2016). Performance evaluation of cloud computing providers using fuzzy multiattribute group decision making model. *2015 12th International Conference on Fuzzy Systems and Knowledge Discovery, FSKD 2015*, (c), 130–135. <https://doi.org/10.1109/FSKD.2015.7381928>
- Grati, R., Boukadi, K., & Ben-Abdallah, H. (2014). A framework for IaaS-to-SaaS monitoring of BPEL processes in the Cloud: Design and evaluation. *Proceedings of IEEE/ACS International Conference on Computer Systems and Applications, AICCSA, 2014*, 557–564. <https://doi.org/10.1109/AICCSA.2014.7073248>
- Grbac, T., & Huljenić, D. (2011). *Product-Focused Software Process Improvement*. (D. Caivano, M. Oivo, M. T. Baldassarre, & G. Visaggio, Eds.), *Product-Focused Software Process Improvement* (Vol. 6759). Berlin, Heidelberg: Springer Berlin Heidelberg. <https://doi.org/10.1007/978-3-642-21843-9>
- Guerrero, C. A., & Londoño, J. M. (2016). Revisión de la Problemática de la Calidad del Software para el Desarrollo de Aplicaciones de Computación en la Nube. *Información tecnológica*, 27(3), 61–80. <https://doi.org/10.4067/S0718-07642016000300007>

- Guha, R., & Al-Dabass, D. (2010). Impact of Web 2.0 and Cloud Computing Platform on Software Engineering. En *2010 International Symposium on Electronic System Design* (pp. 213–218). IEEE. <https://doi.org/10.1109/ISED.2010.48>
- Guo, J. (2003). An approach for modeling and designing software architecture. En *10th IEEE International Conference and Workshop on the Engineering of Computer-Based Systems, 2003. Proceedings.* (pp. 89–97). IEEE Comput. Soc. <https://doi.org/10.1109/ECBS.2003.1194787>
- Halabi, T., & Bellaiche, M. (2018). A broker-based framework for standardization and management of Cloud Security-SLAs. *Computers and Security, 75*, 59–71. <https://doi.org/10.1016/j.cose.2018.01.019>
- Hammam, A., & Senbel, S. (2013). A trust management system for ad-hoc mobile clouds. En *2013 8th International Conference on Computer Engineering & Systems (ICCES)* (pp. 31–38). IEEE. <https://doi.org/10.1109/ICCES.2013.6707167>
- Hashmi, S. I., Clerc, V., Razavian, M., Manteli, C., Tamburri, D. A., Lago, P., ... Richardson, I. (2011). Using the Cloud to Facilitate Global Software Development Challenges. En *2011 IEEE Sixth International Conference on Global Software Engineering Workshop* (pp. 70–77). IEEE. <https://doi.org/10.1109/ICGSE-W.2011.19>
- Hedden, H. (2010). *The accidental taxonomist*. (I. Information Today, Ed.). Medford: Information Today, Inc.
- Hernández-Sampieri, R., Fernández-Collado, C., & Baptista-Lucio, P. (2014). *Metodología de la investigación*. (E. M. Rocha, Ed.) (5a ed.). McGraw-Hill / Interamericana Editores, S.A. de C.V.
- Hershey, P., Rao, S., Silio, C. B., & Narayan, A. (2012). System of Systems to provide Quality of Service monitoring, management and response in cloud computing environments. En *2012 7th International Conference on System of Systems Engineering (SoSE)* (pp. 314–320). IEEE. <https://doi.org/10.1109/SYSoSE.2012.6384208>
- Hlava, M. M. K. (2014). The Taxobook: Principles and Practices of Building Taxonomies: Part 2 of a 3-Part Series. *Synthesis Lectures on Information Concepts, Retrieval, and Services, 6(4)*, 1–164. <https://doi.org/10.2200/S00603ED1V02Y201410ICR036>
- Hoang, D. T., Niyato, D., & Wang, P. (2012). Optimal admission control policy for mobile cloud computing hotspot with cloudlet. En *2012 IEEE Wireless Communications and Networking Conference (WCNC)* (pp. 3145–3149). IEEE. <https://doi.org/10.1109/WCNC.2012.6214347>
- Hobfeld, T., Schatz, R., Varela, M., & Timmerer, C. (2012). Challenges of QoE management for cloud applications. *IEEE Communications Magazine, 50(4)*, 28–36. <https://doi.org/10.1109/MCOM.2012.6178831>
- Hoefler, C. N., & Karagiannis, G. (2010). Taxonomy of cloud computing services. En *2010*

IEEE Globecom Workshops (pp. 1345–1350). IEEE.
<https://doi.org/10.1109/GLOCOMW.2010.5700157>

Hoefler, C. N., & Karagiannis, G. (2011). Cloud computing services: taxonomy and comparison. *Journal of Internet Services and Applications*, 81–94.
<https://doi.org/10.1007/s13174-011-0027-x>

Hongzhen, X., Limin, L., Dehua, X., & Yanqin, L. (2016). Evolution of service composition based on QoS under the cloud computing environment. *2016 IEEE International Conference of Online Analysis and Computing Science (ICOACS), 2016*, 66–69.
<https://doi.org/10.1109/ICOACS.2016.7563050>

Hou, Z., & Zhou, X. (2010). ASAAS: Application Software as a Service for High Performance Cloud Computing. En *2010 IEEE 12th International Conference on High Performance Computing and Communications (HPCC)* (pp. 156–163). IEEE.
<https://doi.org/10.1109/HPCC.2010.76>

Hsueh, N.-L., Wang, J.-Y., & Ting, D.-H. (2012). A Cloud Service Design for Quality Evaluation of Design Patterns. En *2012 Sixth International Conference on Genetic and Evolutionary Computing* (pp. 184–187). IEEE.
<https://doi.org/10.1109/ICGEC.2012.9>

Hu, H., & Zhang, J. (2013). The Evaluation System for Cloud Service Quality Based on SERVQUAL. En W. Lu, G. Cai, W. Liu, & W. Xing (Eds.), *Proceedings of the 2012 International Conference on Information Technology and Software Engineering* (Vol. 210, pp. 577–584). Berlin, Heidelberg: Springer Berlin Heidelberg.
<https://doi.org/10.1007/978-3-642-34528-9>

Huang, C., Hsu, P., & Tzeng, G. (2012). Evaluating cloud computing based telecommunications service quality enhancement by using a new hybrid MCDM model. *Intelligent Decision Technologies*, 1, 519–536. https://doi.org/10.1007/978-3-642-29977-3_52

Huang, J., & Yanbing, L. (2013). Modeling and algorithms for QoS-aware service composition in virtualization-based cloud computing. *IEICE Transactions and Communications*, E96-B(1), 1–10. Recuperado de http://search.ieice.org/bin/summary.php?id=e96-b_1_10

Hussain, M., & Abdulsalam, H. M. (2014). Software quality in the clouds: a cloud-based solution. *Cluster Computing*, 17(2), 389–402. <https://doi.org/10.1007/s10586-012-0233-8>

IEC. (2012). IEC PAS 62814:2012. Dependability of software products containing reusable components - Guidance for functionality and tests. Recuperado el 6 de marzo de 2018, de <https://webstore.iec.ch/publication/7571>

IEC. (2018). International Electrotechnical Commission. Recuperado el 6 de marzo de 2018, de <http://www.iec.ch/>

- IEEE. (1990). IEEE Standard Glossary of Software Engineering Terminology. *IEEE Std 610.12-1990*, 1–84. <https://doi.org/10.1109/IEEESTD.1990.101064>
- IEEE. (1994). IEEE Std 1228-1994 - Standard for Software Safety Plans. <https://doi.org/10.1109/IEEESTD.1994.122165>
- IEEE. (1998). IEEE Std 1061-1998 - Standard for a Software Quality Metrics Methodology. <https://doi.org/10.1109/IEEESTD.1998.243394>
- IEEE. (2006). IEEE Std 982.1-2005 - Standard Dictionary of Measures of the Software Aspects of Dependability, 1–34. <https://doi.org/10.1109/IEEESTD.2006.215280>
- IEEE. (2008). IEEE Std 1028-2008 - Standard for Software Reviews and Audits. <https://doi.org/10.1109/IEEESTD.2008.4601584>
- IEEE. (2010). IEEE Std 1044-2009 - Standard Classification for Software Anomalies. <https://doi.org/10.1109/IEEESTD.2010.5399061>
- IEEE. (2014). IEEE Std 730-2014 - Standard for Software Quality Assurance, 1–138. <https://doi.org/10.1109/IEEESTD.2014.6835311>
- IEEE. (2017). IEEE Std 1012-2016 - Standard for System, Software, and Hardware Verification and Validation, 1–260. <https://doi.org/10.1109/IEEESTD.2017.8055462>
- IEEE. (2018). Institute of Electrical and Electronics Engineers. Recuperado el 6 de marzo de 2018, de <https://www.ieee.org/>
- IGI-Global. (2018). IGI Global. Disseminator of knowledge. Recuperado de <https://www.igi-global.com/dictionary/modeling-content-quality-web-follow/24281>
- Isa, D. I. P., Brogi, A., & Forti, S. (2016). QoS-aware Deployment of IoT Applications Through the Fog, *4*(5), 1185–1192.
- ISO/IEC/IEEE. (2013). ISO/IEC/IEEE 29119 Software Testing. Recuperado de <http://www.softwaretestingstandard.org/>
- ISO/IEC. (2008). ISO/IEC 25012:2008 Software engineering -- Software product Quality Requirements and Evaluation (SQuaRE) - Data quality model, 1–13.
- ISO/IEC. (2011). ISO/IEC 25010:2011 Systems and software engineering -- Systems and software Quality Requirements and Evaluation (SQuaRE) - System and software quality models, 1–34.
- ISO/IEC. (2014). ISO/IEC 25000:2014 - Systems and software engineering -- Systems and software Quality Requirements and Evaluation (SQuaRE) - Guide to SQuaRE, 1–27.
- ISO. (2015). *Norma Internacional ISO 9000-2015*.
- ISO. (2018). International Organization for Standardization. Recuperado el 6 de marzo de

2018, de <https://www.iso.org/home.html>

- ITU. (2018). International Telecommunication Union. Recuperado el 6 de marzo de 2018, de <https://www.itu.int/en/about/Pages/default.aspx>
- Jelassi, M., Ghazel, C., & Saidane, L. A. (2017). A survey on quality of service in cloud computing. En *2017 3rd International Conference on Frontiers of Signal Processing (ICFSP)* (Vol. 4, pp. 63–67). IEEE. <https://doi.org/10.1109/ICFSP.2017.8097142>
- Jhavar, R., & Piuri, V. (2013). Adaptive Resource Management for Balancing Availability and Performance in Cloud Computing. *Proceedings of the 10th International Conference on Security and Cryptography (SECRYPT 2013)*, 254–264. <https://doi.org/10.1128/JVI.77.9.5439>
- Jiménez-Domingo, E. (2013). *Modelo de Interoperabilidad para Plataformas de Cloud Computing basado en Tecnologías del Conocimiento*. Universidad Carlos III de Madrid. Recuperado de http://e-archivo.uc3m.es/bitstream/handle/10016/18172/tesis_enrique_jimenez_domingo_2013.pdf?sequence=1
- Jongkol Janruang, S. G. (2011). Semantic Suffix Tree Clustering. En *First IRAST International Conference on Data Engineering and Internet Technology (DEIT)* (pp. 35–40). <https://doi.org/10.1.1.185.705>
- Kara, M., Lamouchi, O., & Ramdane-Cherif, A. (2017). Semantically equivalent model for quality evaluation. *Proceedings of the Second International Conference on Internet of things and Cloud Computing - ICC '17, 02*, 1–5. <https://doi.org/10.1145/3018896.3056776>
- Karim, R., Ding, C., & Miri, A. (2013). An end-to-end QoS mapping approach for cloud service selection. *Proceedings - 2013 IEEE 9th World Congress on Services, SERVICES 2013*, 341–348. <https://doi.org/10.1109/SERVICES.2013.71>
- Khaddaj, S. (2012). Cloud computing: Service provisioning and user requirements. *Proceedings - 11th International Symposium on Distributed Computing and Applications to Business, Engineering and Science, DCABES 2012*, 191–195. <https://doi.org/10.1109/DCABES.2012.76>
- Khan, H. M., Chan, G. Y., & Chua, F. F. (2016). An adaptive monitoring framework for ensuring accountability and quality of services in cloud computing. *International Conference on Information Networking, 2016-March*, 249–253. <https://doi.org/10.1109/ICOIN.2016.7427071>
- Khan, M. Z., & Qamar, U. (2015). Towards Service Evaluation and Ranking Model for Cloud Infrastructure Selection. En *2015 IEEE 12th Intl Conf on Ubiquitous Intelligence and Computing and 2015 IEEE 12th Intl Conf on Autonomic and Trusted Computing and 2015 IEEE 15th Intl Conf on Scalable Computing and Communications and Its Associated Workshops (UIC-ATC-ScalCom)* (pp. 1282–1287). IEEE. <https://doi.org/10.1109/UIC-ATC-ScalCom-CBDCCom-IoP.2015.232>

- Kim, J., Mohan, K., & Ramesh, B. (2014). Functional and nonfunctional quality in cloud-based collaborative writing: An empirical investigation. *IEEE Transactions on Professional Communication*, 57(3), 182–203. <https://doi.org/10.1109/TPC.2014.2344331>
- Kiruthika, J., Horgan, G., & Khaddaj, S. (2012). Quality Measurement for Cloud Based E-commerce Applications. En *2012 11th International Symposium on Distributed Computing and Applications to Business, Engineering & Science* (pp. 209–213). IEEE. <https://doi.org/10.1109/DCABES.2012.62>
- Kitchenham, B. (2007). Guidelines for performing Systematic Literature Reviews in Software Engineering, *Version 2.*, 65. Recuperado de <http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=F30B95674BEB7E2538E6BCBD8F927EDC?doi=10.1.1.117.471&rep=rep1&type=pdf>
- Klems, M., Bermbach, D., & Weinert, R. (2012). A Runtime Quality Measurement Framework for Cloud Database Service Systems. En *2012 Eighth International Conference on the Quality of Information and Communications Technology* (pp. 38–46). IEEE. <https://doi.org/10.1109/QUATIC.2012.17>
- Korkontzelos, I., Mu, T., & Ananiadou, S. (2012). ASCOT: a text mining-based web-service for efficient search and assisted creation of clinical trials. *BMC Medical Informatics and Decision Making*, 12(Suppl 1), S3. <https://doi.org/10.1186/1472-6947-12-S1-S3>
- Krebs, R., Momm, C., & Kounev, S. (2013). Metrics and techniques for quantifying performance isolation in cloud environments. *Science of Computer Programming*, 1, 1–19. <https://doi.org/10.1016/j.scico.2013.08.003>
- Kumar, N., Chilamkurti, N., Zeadally, S., & Jeong, Y.-S. (2013). Achieving Quality of Service (QoS) Using Resource Allocation and Adaptive Scheduling in Cloud Computing with Grid Support. *The Computer Journal*, 57(2), 281–290. <https://doi.org/10.1093/comjnl/bxt024>
- Kumar, P., Sehgal, V., & Diwakar, M. (2011). Clouds: Concept to optimize the Quality of Service (QOS) for clusters. En *2011 World Congress on Information and Communication Technologies* (pp. 816–821). IEEE. <https://doi.org/10.1109/WICT.2011.6141352>
- Kumar, R. R., & Kumar, C. (2017). Designing an efficient methodology based on Entropy-TOPSIS for evaluating efficiency of cloud services. *Proceedings of the 7th International Conference on Computer and Communication Technology - ICCCT-2017*, 117–122. <https://doi.org/10.1145/3154979.3154982>
- La, H. J., & Kim, S. D. (2013). A model of quality-in-use for service-based mobile ecosystem. En *2013 1st International Workshop on the Engineering of Mobile-Enabled Systems (MOBS)* (pp. 13–18). IEEE. <https://doi.org/10.1109/MOBS.2013.6614217>
- Lee, H., Kim, J., Lee, Y., & Won, D. (2012). Security Issues and Threats According to the

- Attribute of Cloud Computing (Vol. 339, pp. 101–108). https://doi.org/10.1007/978-3-642-35264-5_14
- Lee, J. Y., Lee, J. W., Cheun, D. W., & Kim, S. D. (2009). A Quality Model for Evaluating Software-as-a-Service in Cloud Computing. En *2009 Seventh ACIS International Conference on Software Engineering Research, Management and Applications* (pp. 261–266). IEEE. <https://doi.org/10.1109/SERA.2009.43>
- Lehrig, S., & Becker, S. (2015). Beyond simulation: Composing scalability, elasticity, and efficiency analyses from preexisting analysis results. *WOSP-C 2015 - Proceedings of the 2015 ACM/SPEC Workshop on Challenges in Performance Methods for Software Development, in Conjunction with ICPE 2015*, 29–34. <https://doi.org/10.1145/2693561.2693568>
- Lehrig, Sebastian, Eikerling, H., & Becker, S. (2015). Scalability , elasticity , and efficiency in cloud computing: a systematic literature review of definitions and metrics. *Proceedings of the 11th International ACM SIGSOFT Conference on Quality of Software Architectures*, (1), 83–92. <https://doi.org/10.1145/2737182.2737185>
- Leiden University. (2018). VOSviewer Visualizing scientific landscapes. Recuperado el 10 de julio de 2018, de <http://www.vosviewer.com/>
- Li, X., & Yu, Q. (2011). Design and implementation of quality tracking system based on cloud computing. En *Proceedings of 2011 International Conference on Computer Science and Network Technology* (pp. 2495–2498). IEEE. <https://doi.org/10.1109/ICCSNT.2011.6182476>
- Li, Z., OBrien, L., Cai, R., & Zhang, H. (2012). Towards a Taxonomy of Performance Evaluation of Commercial Cloud Services. *2012 IEEE Fifth International Conference on Cloud Computing*, 344–351. <https://doi.org/10.1109/CLOUD.2012.74>
- Lim, E., & Thiran, P. (2014). Communication of technical QoS among cloud brokers. *Proceedings - 2014 IEEE International Conference on Cloud Engineering, IC2E 2014*, 403–409. <https://doi.org/10.1109/IC2E.2014.92>
- Lin, S. Y., Lai, C. H., Wu, C. H., & Lo, C. C. (2014). A trustworthy QoS-based collaborative filtering approach for web service discovery. *Journal of Systems and Software*, 93, 217–228. <https://doi.org/10.1016/j.jss.2014.01.036>
- Liu, H., Bu, F., & Cai, H. (2012). SLA-based service composition model with semantic support. *Proceedings - 2012 IEEE Asia-Pacific Services Computing Conference, APSCC 2012*, 374–379. <https://doi.org/10.1109/APSCC.2012.54>
- Liu, Z., Liu, T., Lu, T., Cai, L., & Yang, G. (2010). Agent-Based Online Quality Measurement Approach in Cloud Computing Environment. En *2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology* (pp. 686–690). IEEE. <https://doi.org/10.1109/WI-IAT.2010.213>
- Lu, L., & Yuan, Y. (2018). A novel TOPSIS evaluation scheme for cloud service

- trustworthiness combining objective and subjective aspects. *Journal of Systems and Software*, 143(April), 71–86. <https://doi.org/10.1016/j.jss.2018.05.004>
- Ma, Z., Jiang, R., Yang, M., Li, T., & Zhang, Q. (2018). Research on the measurement and evaluation of trusted cloud service. *Soft Computing*, 22(4), 1247–1262. <https://doi.org/10.1007/s00500-016-2412-7>
- Mantecón, A., Juan, M., Calafat, A., Becoña, E., & Román, E. (2008). Respondent-Driven Sampling: un nuevo método de muestreo para el estudio de poblaciones visibles y ocultas. *Adicciones*, 161–170. Recuperado de <http://adicciones.es/index.php/adicciones/article/view/280>
- Marinescu, C., & Petcu, D. (2013). Quality Assessment in the Cloud: Is It Worthwhile? En *2013 17th European Conference on Software Maintenance and Reengineering* (pp. 453–456). IEEE. <https://doi.org/10.1109/CSMR.2013.70>
- Massonet, P., & Arenas, A. (2012). Towards more Transparency in Quality of Protection for Infrastructure Clouds. En *2012 Eighth International Conference on the Quality of Information and Communications Technology* (pp. 285–290). IEEE. <https://doi.org/10.1109/QUATIC.2012.62>
- Medeiros, N. P. D. S., Ivaki, N. R., Costa, P. N. Da, & Vieira, M. P. A. (2017). Towards an Approach for Trustworthiness Assessment of Software as a Service. *Proceedings - 2017 IEEE 1st International Conference on Edge Computing, EDGE 2017*, 220–223. <https://doi.org/10.1109/IEEE.EDGE.2017.39>
- Medianero-Burga, D. (2011). Metodología de Estudios de Línea de Base. *Pensamiento crítico*, 15, 61–82. Recuperado de <http://revistasinvestigacion.unmsm.edu.pe/index.php/econo/article/view/8994/7822>
- Mell, P., & Grance, T. (2011). The NIST Definition of Cloud Computing-Recommendations of the National Institute of Standards and Technology. NIST. Recuperado de <http://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:The+NIST+Definition+of+Cloud+Computing+Recommendations+of+the+National+Institute+of+Standards+and+Technology#8>
- Meyer, A., Ramirez, L., & Pérez, C. (2013). Percepción de estrés en estudiantes chilenos de Medicina y Enfermería. *Educ Cienc Salud*, 10(2), 79–85. Recuperado de [http://repositoriodigital.ucsc.cl/bitstream/handle/25022009/360/Andrea Meyer%2CLuis Ramirez%2C Cristhian Pérez.pdf?sequence=1&isAllowed=y](http://repositoriodigital.ucsc.cl/bitstream/handle/25022009/360/Andrea_Meyer%2CLuis_Ramirez%2C_Cristhian_Perez.pdf?sequence=1&isAllowed=y)
- Miguel, R.-G. C., Alessandro, D. B.-S. G., & David, C.-C. A. (2016). Conocimientos, actitudes y prácticas sobre el Zika en estudiantes de medicina, 2016. *Original*, 19(2), 33–37. Recuperado de http://www.scielo.org/bo/pdf/rccm/v19n2/v19n2_a06.pdf
- Miyamoto, K., Nerome, T., & Nakamura, T. (2012). Document Quality Checking Tool for Global Software Development. En *2012 Annual SRII Global Conference* (pp. 267–276). IEEE. <https://doi.org/10.1109/SRII.2012.37>

- Moghaddam, F. F., Majd, A., Ahmadi, M., Khodadadi, T., & Madadipouya, K. (2016). A dynamic classification index to enhance data protection procedures in cloud-based environments. *Proceedings - 2015 6th IEEE Control and System Graduate Research Colloquium, ICSGRC 2015*, 17–22. <https://doi.org/10.1109/ICSGRC.2015.7412457>
- Mohamed, M. F. (2016). Service replication taxonomy in distributed environments. *Service Oriented Computing and Applications*, 10(3), 317–336. <https://doi.org/10.1007/s11761-015-0189-7>
- Mohan, K., & Aramudhan, M. (2017). Broker based trust architecture for federated healthcare cloud system. *Intelligent Automation & Soft Computing*, 23(3), 477–483. <https://doi.org/10.1080/10798587.2016.1220118>
- Monir, M. B., AbdelAziz, M. H., AbdelHamid, A. A., & El-Horbaty, E.-S. M. (2015). Trust management in cloud computing: A survey. En *2015 IEEE Seventh International Conference on Intelligent Computing and Information Systems (ICICIS)* (pp. 231–242). IEEE. <https://doi.org/10.1109/IntelCIS.2015.7397227>
- Motta, G., You, L., Sacco, D., & Sfondrini, N. (2013). Cloud computing: The issue of service quality: An overview of cloud service level management architectures. *Proceedings - 2013 5th International Conference on Service Science and Innovation, ICSSI 2013*, 230–233. <https://doi.org/10.1109/ICSSI.2013.48>
- Mrabet, M., Saied, Y. Ben, & Saidane, L. A. (2017a). A new trust evaluation approach for cloud computing environments. *5th IFIP International Conference on Performance Evaluation and Modeling in Wired and Wireless Networks, PEMWN 2016*. <https://doi.org/10.1109/PEMWN.2016.7842907>
- Mrabet, M., Saied, Y. Ben, & Saidane, L. A. (2017b). Modeling Correlation between QoS Attributes for Trust Computation in Cloud Computing Environments. *Proceedings - 2017 17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing, CCGRID 2017*, 488–497. <https://doi.org/10.1109/CCGRID.2017.38>
- Muñoz-Escóí, F. D., & Bernabéu-Aubán, J. M. (2017). A survey on elasticity management in PaaS systems. *Computing*, 99(7), 617–656. <https://doi.org/10.1007/s00607-016-0507-8>
- Mustafa, S., Nazir, B., Hayat, A., Khan, A. ur R., & Madani, S. A. (2015). Resource management in cloud computing: Taxonomy, prospects, and challenges. *Computers & Electrical Engineering*, 47, 186–203. <https://doi.org/10.1016/j.compeleceng.2015.07.021>
- Nabi, M., Toeroe, M., & Khendek, F. (2016). Availability in the cloud: State of the art. *Journal of Network and Computer Applications*, 60, 54–67. <https://doi.org/10.1016/j.jnca.2015.11.014>
- Nadanam, P., & Rajmohan, R. (2012). QoS evaluation for web services in cloud computing. En *2012 3rd International Conference on Computing, Communication and Networking Technologies, ICCCNT 2012* (pp. 1–8). IEEE.

<https://doi.org/10.1109/ICCCNT.2012.6395991>

- Naseer, M. K., Jabbar, S., & Zafar, I. (2014). A novel trust model for selection of Cloud Service Provider. *2014 World Symposium on Computer Applications and Research, WSCAR 2014*. <https://doi.org/10.1109/WSCAR.2014.6916772>
- Naseer, M., & Nazar, M. (2017). A framework for selection of SAAS by evaluating the quality of Freemium model. *2016 6th International Conference on Innovative Computing Technology, INTECH 2016*, 78–82. <https://doi.org/10.1109/INTECH.2016.7845029>
- NIST. (2011a). *Cloud Architecture Reference Models: A survey*. Estados Unidos. Recuperado de <https://www.ogf.org/pipermail/occiwg/attachments/20110128/fe1a4498/attachment-0001.pdf>
- NIST. (2011b). *NIST Cloud Computing Reference Architecture*. Estados Unidos.
- Nuñez, D., Fernández-Gago, C., & Luna, J. (2016). Eliciting metrics for accountability of cloud systems. *Computers and Security*, 62, 149–164. <https://doi.org/10.1016/j.cose.2016.07.003>
- OCDE - Eurostat. (2007). *Manual de Oslo: Directrices para la recogida de información e interpretación de información relativa e innovación*. Recuperado de <http://www.madrid.org/bvirtual/BVCM001708.pdf>
- Oestreich, K. (2013). Converged Infrastructure. Recuperado el 8 de mayo de 2017, de <http://www.cioandleader.com/articles/7505/converged-infrastructure>
- Oh, S. H., La, H. J., & Kim, S. D. (2011). A reusability evaluation suite for cloud services. *Proceedings - 2011 8th IEEE International Conference on e-Business Engineering, ICEBE 2011*, 111–118. <https://doi.org/10.1109/ICEBE.2011.27>
- Olokunde, T., Misra, S., & Adewumi, A. (2017). Quality Model for Evaluating Platform as a Service in Cloud Computing. En T. Skersys, R. Butleris, & R. Butkiene (Eds.), *Information and Software Technologies* (Vol. 319, pp. 280–291). Berlin, Heidelberg, Heidelberg: Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-319-67642-5_23
- OpenCrowd. (2010). Cloud Taxonomy. Recuperado el 8 de junio de 2017, de <http://cloudtaxonomy.opencrowd.com/>
- Osiński, S., Stefanowski, J., & Weiss, D. (2004). Lingo: Search Results Clustering Algorithm Based on Singular Value Decomposition. En *Intelligent Information Processing and Web Mining* (pp. 359–368). Berlin, Heidelberg: Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-540-39985-8_37
- P. Miguel, J., Mauricio, D., & Rodríguez, G. (2014). A Review of Software Quality Models for the Evaluation of Software Products. *International Journal of Software Engineering & Applications*, 5(6), 31–53. <https://doi.org/10.5121/ijsea.2014.5603>

- Padmapriya, N., & Rajmohan, R. (2012). Reliability evaluation suite for cloud services. En *2012 Third International Conference on Computing, Communication and Networking Technologies (ICCCNT'12)* (Vol. 115, pp. 1–6). IEEE. <https://doi.org/10.1109/ICCCNT.2012.6395992>
- Pandey, S., & Daniel, A. K. (2016). Fuzzy logic based cloud service trustworthiness model. *Proceedings of 2nd IEEE International Conference on Engineering and Technology, ICETECH 2016*, (March), 73–78. <https://doi.org/10.1109/ICETECH.2016.7569215>
- Patidar, S., Rane, D., & Jain, P. (2011). Challenges of software development on cloud platform. En *2011 World Congress on Information and Communication Technologies* (pp. 1009–1013). IEEE. <https://doi.org/10.1109/WICT.2011.6141386>
- Pawluk, P., Litoiu, M., & Cercone, N. (2011). FROM QoD TO QoS - Data Quality Issues in Cloud Computing. En *Proceedings of the 1st International Conference on Cloud Computing and Services Science* (pp. 697–702). SciTePress - Science and Technology Publications. <https://doi.org/10.5220/0003558606970702>
- Pedersen, J. M., Riaz, M. T., Celestino, J., Dubalski, B., Ledzinski, D., & Patel, A. (2011). Assessing measurements of QoS for global cloud computing services. *Proceedings - IEEE 9th International Conference on Dependable, Autonomic and Secure Computing, DASC 2011*, 682–689. <https://doi.org/10.1109/DASC.2011.120>
- Perianes-Rodriguez, A., Waltman, L., & Van Eck, N. J. (2016). Constructing bibliometric networks: A comparison between full and fractional counting. *Journal of Informetrics*, 10(4), 1178–1195. <https://doi.org/10.1016/j.joi.2016.10.006>
- Petticrew, M., & Roberts, H. (2005). *Systematic Reviews in the Social Sciences: A Practical Guide*.
- Phani Sheetal, A., & Ravindranath, K. (2018). Software metric evaluation on cloud based applications. *International Journal of Engineering & Technology*, 7(1.5 Special Issue 5), 13–18. Recuperado de <https://www.scopus.com/record/display.uri?eid=2-s2.0-85040714532&origin=resultslist&sort=plf-f&src=s&nlo=&nlr=&nls=&imp=t&sid=f6a71ac78ae35ac02a691ec72221e418&sot=br&sdt=sisr&sl=62&s=SOURCE-ID%2821100805731%29+AND+PUBYEAR+IS+2018+AND+NOT+DOCTYPE%28ip%29&r>
- Pintos, J. M., Castillo, C. N., & Lopez-Pires, F. (2016). Evaluation and comparison framework for platform as a service providers. En *2016 XLII Latin American Computing Conference (CLEI)* (pp. 1–11). IEEE. <https://doi.org/10.1109/CLEI.2016.7833384>
- Pizard, S. (2016). *Hacia una taxonomía sobre educación de ingeniería de software*. Montevideo, Uruguay. Recuperado de <https://www.fing.edu.uy/biblio/hacia-una-taxonomia-de-educacion-de-ingenieria-de-software>
- Polash, F., Abuhussein, A., & Shiva, S. (2014). A Survey of Cloud Computing

Taxonomies : Rationale and Overview, (1), 459–465.

- Qian, H., Medhi, D., & Trivedi, K. (2011). A hierarchical model to evaluate quality of experience of online services hosted by cloud computing. En *12th IFIP/IEEE International Symposium on Integrated Network Management (IM 2011) and Workshops* (pp. 105–112). IEEE. <https://doi.org/10.1109/INM.2011.5990680>
- Qu, C., & Buyya, R. (2014). A cloud trust evaluation system using hierarchical fuzzy inference system for service selection. *Proceedings - International Conference on Advanced Information Networking and Applications, AINA*, 850–857. <https://doi.org/10.1109/AINA.2014.104>
- RĂDULESCU, C. Z., & RĂDULESCU, I. C. (2017). An Extended TOPSIS Approach for Ranking Cloud Service Providers. *Studies in Informatics and Control*, 26(2). <https://doi.org/10.24846/v26i2y201706>
- Raj, G., Sarfaraz, M., & Singh, D. (2014). Survey on trust establishment in cloud computing. *Proceedings of the 5th International Conference on Confluence 2014: The Next Generation Information Technology Summit*, 215–220. <https://doi.org/10.1109/CONFLUENCE.2014.6949375>
- Raj, P., Venkatesh, V., & Amirtharajan, R. (2013). Envisioning the Cloud-Induced Transformations in the Software Engineering Discipline. En Z. Mahmood & S. Saeed (Eds.), *Software Engineering Frameworks for the Cloud Computing Paradigm* (pp. 25–53). London: Springer London. <https://doi.org/10.1007/978-1-4471-5031-2>
- Rajeswari, S., & Kalaiselvi, R. (2018). Survey of data and storage security in cloud computing. *IEEE International Conference on Circuits and Systems, ICCS 2017, 2018-Janua(lccs)*, 76–81. <https://doi.org/10.1109/ICCS1.2017.8325966>
- Ran, Y., Yang, J., Zhang, S., & Xi, H. (2017). Dynamic IaaS computing resource provisioning strategy with QoS constraint. *IEEE Transactions on Services Computing*, 10(2), 190–202. <https://doi.org/10.1109/TSC.2015.2464212>
- Ré, R., Meloca, R. M., Roma, D. N., Ismael, M. A. da C., & Silva, G. C. (2018). An empirical study for evaluating the performance of multi-cloud APIs. *Future Generation Computer Systems*, 79, 726–738. <https://doi.org/10.1016/j.future.2017.09.003>
- Recio-García, J. A., Díaz-Agudo, B., & González-Calero, P. A. (2008). Textual CBR in jCOLIBRI: From Retrieval to Reuse.
- Redmond-Neal, A. (2013). *Starting a Taxonomy Project: Taxonomy Basics*. Recuperado de https://www.sla.org/wp-content/uploads/2013/05/StartingTaxProject_Redmond-Neal.pdf
- Reis, L. P., Costa, A. P., & de Souza, F. N. (2016). A survey on computer assisted qualitative data analysis software. En *2016 11th Iberian Conference on Information Systems and Technologies (CISTI)* (pp. 1–6). IEEE. <https://doi.org/10.1109/CISTI.2016.7521502>

- Reynolds, M. B., Hopkinson, K. M., Oxley, M. E., & Mullins, B. E. (2011). Provisioning Norm: An Asymmetric Quality Measure for SaaS Resource Allocation. En *2011 IEEE International Conference on Services Computing* (pp. 112–119). IEEE. <https://doi.org/10.1109/SCC.2011.16>
- Richardson, I., Casey, V., McCaffery, F., Burton, J., & Beecham, S. (2012). A Process Framework for Global Software Engineering Teams. *Information and Software Technology*, *54*(11), 1175–1191. <https://doi.org/10.1016/j.infsof.2012.05.002>
- Rimal, B., Choi, E., & Lumb, I. (2009). A Taxonomy and Survey of Cloud Computing Systems. *2009 Fifth International Joint Conference on INC, IMS and IDC*, 44–51. <https://doi.org/10.1109/NCM.2009.218>
- Ritu, & Jain, S. (2017). A trust model in cloud computing based on fuzzy logic. *2016 IEEE International Conference on Recent Trends in Electronics, Information and Communication Technology, RTEICT 2016 - Proceedings*, 47–52. <https://doi.org/10.1109/RTEICT.2016.7807780>
- Ruby, A. J., Aisha, B. W., & Subash, C. P. (2016). RenderSelect: a Cloud Broker Framework for Cloud Renderfarm Services. Recuperado de <http://arxiv.org/abs/1611.10210>
- Sagbo, K., & Houngue, P. (2012). Quality architecture for resource allocation in cloud computing. *Service-Oriented and Cloud Computing*, 154–168. https://doi.org/10.1007/978-3-642-33427-6_11
- Salama, M., Shawish, A., Zeid, A., & Kouta, M. (2012). Integrated QoS utility-based model for cloud computing service provider selection. *Proceedings - International Computer Software and Applications Conference*, 45–50. <https://doi.org/10.1109/COMPSACW.2012.18>
- Salger, F., Engels, G., & Hofmann, A. (2010). Assessments in global software development. En *Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering - ICSE '10* (Vol. 2, p. 29). New York, New York, USA: ACM Press. <https://doi.org/10.1145/1810295.1810301>
- Salleh, M. A., Bahari, M., & Zakaria, N. H. (2017). An Overview of Software Functionality Service: A Systematic Literature Review. *Procedia Computer Science*, *124*, 337–344. <https://doi.org/10.1016/j.procs.2017.12.163>
- Saravanan, M., Aramudhan, M., Sundara Pandiyan, S., & Avudaiappan, T. (2018). Priority based prediction mechanism for ranking providers in federated cloud architecture. *Cluster Computing*. <https://doi.org/10.1007/s10586-017-1593-x>
- Shin, Y. R., & Huh, E. N. (2015). QoE metrics aggregation for hierarchical Service Level Agreement in Cross-Layered SLA architecture. *International Conference on Ubiquitous and Future Networks, ICUFN, 2015-Augus*, 831–836. <https://doi.org/10.1109/ICUFN.2015.7182659>

- Shin, Y. R., & Huh, E. N. (2016). MCSQAM: Service Quality Assessment Model in Mobile Cloud Services Environment. *Mobile Information Systems*, 2016(1). <https://doi.org/10.1155/2016/2517052>
- Shu, Y., Zhao, Y., Liu, H., Zuo, D., & Yang, X. (2015). A Hybrid QoS Evaluation Tool Based on the Cloud Computing Platform (pp. 822–835). https://doi.org/10.1007/978-3-319-27140-8_57
- Siegel, J., & Perdue, J. (2012). Cloud Services Measures for Global Use: The Service Measurement Index (SMI). En *2012 Annual SRII Global Conference* (pp. 411–415). IEEE. <https://doi.org/10.1109/SRII.2012.51>
- Singh, P. K., Suryawanshi, A., Gupta, S., & Saindane, P. (2016). Elasticsearch and Carrot2-Based Log Analytics and Management (pp. 71–78). https://doi.org/10.1007/978-981-10-0419-3_9
- Singh, S., & Chana, I. (2012). Cloud Based Development Issues: A Methodical Analysis. *International Journal of Cloud Computing and Services Science (IJ-CLOSER)*, 2(1), 73–84. <https://doi.org/10.11591/closer.v2i1.1704>
- Singh, S., & Chana, I. (2015). Q-Aware: Quality of service based cloud resource provisioning. *Computers and Electrical Engineering*, 47, 138–160. <https://doi.org/10.1016/j.compeleceng.2015.02.003>
- Sivaraj, S., & Muniraj, N. J. R. (2014). Efficient QOS with multiple constraints for epigenomic scientific workflow model in SAAS cloud. *Journal of Theoretical and Applied Information Technology*, 65(1), 242–247.
- Smith, D., Leong, L., & Bala, R. (2018). *Magic Quadrant for Cloud Infrastructure as a Service, Worldwide*. Recuperado de <https://www.gartner.com/doc/reprints?id=1-50WJ5CK&ct=180525&st=sb>
- Sodhi, B., & Prabhakar, T. V. (2012). Assessing platform suitability for achieving quality in guest applications. *Proceedings - Asia-Pacific Software Engineering Conference, APSEC, 1*, 760–765. <https://doi.org/10.1109/APSEC.2012.53>
- Somu, N., Kirthivasan, K., & V.S., S. S. (2017). A computational model for ranking cloud service providers using hypergraph based techniques. *Future Generation Computer Systems*, 68, 14–30. <https://doi.org/10.1016/j.future.2016.08.014>
- Song, J., Zhang, S., Gong, Y., & Dai, B. (2012). A QoS Evaluation Model for Test-Bed in the Cloud Computing Environment. En *2012 IEEE Ninth International Conference on e-Business Engineering* (pp. 292–295). IEEE. <https://doi.org/10.1109/ICEBE.2012.54>
- Sousa, F. R. C., Moreira, L. O., Santos, G. A. C., & Machado, J. C. (2012). Quality of service for database in the cloud. *CLOSER 2012 - Proceedings of the 2nd International Conference on Cloud Computing and Services Science*, 595–601. <https://doi.org/10.1007/s00256-011-1257-5>

- Sun, D., Chang, G., Miao, C., & Wang, X. (2013). Analyzing, modeling and evaluating dynamic adaptive fault tolerance strategies in cloud computing environments. *The Journal of Supercomputing*, 66(1), 193–228. <https://doi.org/10.1007/s11227-013-0898-7>
- Tajvidi, M., Ranjan, R., Kolodziej, J., & Wang, L. (2014). Fuzzy cloud service selection framework. En *2014 IEEE 3rd International Conference on Cloud Networking (CloudNet)* (pp. 443–448). IEEE. <https://doi.org/10.1109/CloudNet.2014.6969035>
- Takabi, H. (2013). *A semantic based policy management framework for cloud computing environments*. University of Pittsburgh. Recuperado de <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.671.8234&rep=rep1&type=pdf>
- Talukder, A. K., & Prahalad, H. A. (2009). Security & scalability architecture for next generation internet services. *2009 IEEE International Conference on Internet Multimedia Services Architecture and Applications, IMSAA 2009*. <https://doi.org/10.1109/IMSAA.2009.5439441>
- Tan, Z.-H., Liu, Y.-C., Shi, N.-X., Wang, X.-W., & Guo, N. (2017). MCTModel: A multi-clouds trust model based on SLA in cloud computing. *Journal of Computers (Taiwan)*, 28(6), 236–245. <https://doi.org/10.3966/199115992017122806021>
- Tan, Z., Niu, Y., Liu, Y., & Yang, G. (2016). A novel trust model based on SLA and behavior evaluation for clouds. *2016 14th Annual Conference on Privacy, Security and Trust, PST 2016*, 581–587. <https://doi.org/10.1109/PST.2016.7907020>
- Tang, W., & Yan, Z. (2015). CloudRec: A mobile cloud service recommender system based on adaptive QoS management. *Proceedings - 14th IEEE International Conference on Trust, Security and Privacy in Computing and Communications, TrustCom 2015*, 1, 9–16. <https://doi.org/10.1109/Trustcom.2015.351>
- Tao, Q., Chang, H., Yi, Y., & Gu, C. (2010). A trustworthy management approach for cloud services QoS data. En *2010 International Conference on Machine Learning and Cybernetics* (pp. 1626–1631). IEEE. <https://doi.org/10.1109/ICMLC.2010.5580998>
- Tchernykh, A., Schwiegelsohn, U., Talbi, E. ghazali, & Babenko, M. (2016). Towards understanding uncertainty in cloud computing with risks of confidentiality, integrity, and availability. *Journal of Computational Science*. <https://doi.org/10.1016/j.jocs.2016.11.011>
- Techio, L. R., & Misaghi, M. (2015). EMSCLOUD - An evaluative model of cloud services cloud service management. *5th International Conference on Innovative Computing Technology, INTECH 2015*, (Intech), 100–105. <https://doi.org/10.1109/INTECH.2015.7173479>
- Tolosana-Calasanz, R., Bañares, J. Á., Pham, C., & Rana, O. (2012). End-to-end QoS on shared clouds for highly dynamic, large-scale sensing data streams. *Proceedings - 12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing*,

CCGrid 2012, 904–911. <https://doi.org/10.1109/CCGrid.2012.67>

- Totiya, S., & Senivongse, T. (2017). Framework to support cloud service selection based on service measurement index. *Proceedings of the World Congress on Engineering and Computer Science 2017*, 1.
- Tripathi, A., Pathak, I., & Vidyarthi, D. P. (2017). Integration of analytic network process with service measurement index framework for cloud service provider selection. *Concurrency and Computation: Practice and Experience*, 29(12), e4144. <https://doi.org/10.1002/cpe.4144>
- Trueba-Gómez, R., & Estrada-Lorenzo, J.-M. (2010). La base de datos PubMed y la búsqueda de información científica. *Seminarios de la Fundación Española de Reumatología*, 11(2), 49–63. <https://doi.org/10.1016/j.semreu.2010.02.005>
- Undheim, A., Chilwan, A., & Heegaard, P. (2011). Differentiated availability in cloud computing SLAs. *Proceedings - 2011 12th IEEE/ACM International Conference on Grid Computing, Grid 2011*, 129–136. <https://doi.org/10.1109/Grid.2011.25>
- van Eck, N. J., & Waltman, L. (2010). Software survey: VOSviewer, a computer program for bibliometric mapping. *Scientometrics*, 84(2), 523–538. <https://doi.org/10.1007/s11192-009-0146-3>
- Van Eck, N. J., & Waltman, L. (2009). How to normalize cooccurrence data? An analysis of some well-known similarity measures. *Journal of the American Society for Information Science and Technology*, 60(8), 1635–1651. <https://doi.org/10.1002/asi.21075>
- Van Eck, N. J., & Waltman, L. (2014). Visualizing Bibliometric Networks. En *Measuring Scholarly Impact* (pp. 285–320). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-319-10377-8_13
- Wagle, S. S., Guzek, M., Bouvry, P., & Bisdorff, R. (2016). An evaluation model for selecting cloud services from commercially available cloud providers. *Proceedings - IEEE 7th International Conference on Cloud Computing Technology and Science, CloudCom 2015*, 107–114. <https://doi.org/10.1109/CloudCom.2015.94>
- Wang, Shangguang, Liu, Z., Sun, Q., Zou, H., & Yang, F. (2012). Towards an accurate evaluation of quality of cloud service in service-oriented cloud computing. *Journal of Intelligent Manufacturing*, 25(2), 283–291. <https://doi.org/10.1007/s10845-012-0661-6>
- Wang, Shou-xin, Zhang, L., Wang, S., & Qiu, X. (2010). A Cloud-Based Trust Model for Evaluating Quality of Web Services. *Journal of Computer Science and Technology*, 25(6), 1130–1142. <https://doi.org/10.1007/s11390-010-9394-1>
- Wang, Z. E., & Liu, J. L. (2012). A QoS Evaluation Model in the Environment of Cloud Computing. *Advanced Materials Research*, 488–489, 1094–1100. <https://doi.org/10.4028/www.scientific.net/AMR.488-489.1094>

- Wang, Z., Jiang, N., & Zhou, P. (2015). Quality Model of Maintenance Service for Cloud Computing. En *2015 IEEE 17th International Conference on High Performance Computing and Communications, 2015 IEEE 7th International Symposium on Cyberspace Safety and Security, and 2015 IEEE 12th International Conference on Embedded Software and Systems* (pp. 1460–1465). IEEE. <https://doi.org/10.1109/HPCC-CSS-ICSS.2015.136>
- Wang, Z. Y., Li, Q., Cao, Z. C., Li, W. H., Li, J., & Du, R. Y. (2012). A model-based deployment framework of integrated public cloud service. *Proceedings - 2012 International Conference on Computer Science and Service System, CSSS 2012*, 723–728. <https://doi.org/10.1109/CSSS.2012.186>
- Wen, P. X., & Dong, L. (2013). Quality Model for Evaluating SaaS Service. En *2013 Fourth International Conference on Emerging Intelligent Data and Web Technologies* (pp. 83–87). IEEE. <https://doi.org/10.1109/EIDWT.2013.19>
- Werner, J., Westphall, C. B. C. M., Weingartner, R., Geronimo, G. A., & Westphall, C. B. C. M. (2015). An Approach to IdM with Privacy in the Cloud. *2015 IEEE International Conference on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing*, 168–175. <https://doi.org/10.1109/CIT/IUCC/DASC/PICOM.2015.26>
- White, G., Nallur, V., & Clarke, S. (2017). Quality of service approaches in IoT: A systematic mapping. *Journal of Systems and Software*, 132, 186–203. <https://doi.org/10.1016/j.jss.2017.05.125>
- Xiao, Z., & Xiao, Y. (2013). Security and Privacy in Cloud Computing. *IEEE Communications Surveys & Tutorials*, 15(2), 843–859. <https://doi.org/10.1109/SURV.2012.060912.00182>
- Yadav, N., & Goraya, M. S. (2018). Two-way Ranking Based Service Mapping in Cloud Environment. *Future Generation Computer Systems*, 81, 53–66. <https://doi.org/10.1016/j.future.2017.11.027>
- Yang, K., & Jia, X. (2011). Data storage auditing service in cloud computing: challenges, methods and opportunities. *World Wide Web*, 15(4), 409–428. <https://doi.org/10.1007/s11280-011-0138-0>
- Yaqub, E., Yahyapour, R., Wieder, P., Kotsokalis, C., Lu, K., & Jehangiri, A. I. (2014). Optimal negotiation of service level agreements for cloud-based services through autonomous agents. *Proceedings - 2014 IEEE International Conference on Services Computing, SCC 2014*, 59–66. <https://doi.org/10.1109/SCC.2014.17>
- You, X., Wan, J., Xu, X., Jiang, C., Zhang, W., & Zhang, J. (2011). ARAS-M: Automatic Resource Allocation Strategy based on Market Mechanism in Cloud Computing. *Journal of Computers*, 6(7), 1287–1296. <https://doi.org/10.4304/jcp.6.7.1287-1296>
- Zhang, P., Han, Q., Li, W., Leung, H., & Song, W. (2016). A novel QoS prediction

- approach for cloud service based on Bayesian networks model. *Proceedings - 2016 IEEE International Conference on Mobile Services, MS 2016*, 111–118. <https://doi.org/10.1109/MobServ.2016.26>
- Zhang, P., Zhou, X., Li, W., & Gao, J. (2017). A survey on quality assurance techniques for big data applications. *Proceedings - 3rd IEEE International Conference on Big Data Computing Service and Applications, BigDataService 2017*, 313–319. <https://doi.org/10.1109/BigDataService.2017.42>
- Zheng, X., Martin, P., Brohman, K., & Xu, L. Da. (2014). Cloudqual: A quality model for cloud services. *IEEE Transactions on Industrial Informatics*, 10(2), 1527–1536. <https://doi.org/10.1109/TII.2014.2306329>
- Zheng, Z., Wu, X., Zhang, Y., Lyu, M. R., & Wang, J. (2013). QoS Ranking Prediction for Cloud Services. *IEEE Transactions on Parallel and Distributed Systems*, 24(6), 1213–1222. <https://doi.org/10.1109/TPDS.2012.285>
- Zhou, P., Wang, Z., Li, W., & Jiang, N. (2015). Quality Model of Cloud Service. En *2015 IEEE 17th International Conference on High Performance Computing and Communications, 2015 IEEE 7th International Symposium on Cyberspace Safety and Security, and 2015 IEEE 12th International Conference on Embedded Software and Systems* (pp. 1418–1423). IEEE. <https://doi.org/10.1109/HPCC-CSS-ICCESS.2015.134>
- Zrnec, A., & Lavbič, D. (2011). Comparison of Cloud vs. Tape Backup Performance and Costs with Oracle Database. *Journal of Information and Organizational Sciences*, 35(1), 135–142. Recuperado de <http://hrcak.srce.hr/file/103900>

Apéndice A

Detalle de clústeres generados por Carrot2

Clúster	Clúster principal nivel 1
Consulta	usability security privacy performance financial assurance agility accountability quality attrib metric
Nodos	<div style="display: flex; flex-wrap: wrap;"> <div style="width: 50%;"> <ul style="list-style-type: none"> 📁 Evaluation Model (20) 📁 Reliability (19) → SMI 📁 Availability (17) → SMI 📁 Service Provider (17) 📁 Service Selection (16) 📁 Quality Attributes (15) 📁 Measurement (12) 📁 Metric Model (12) 📁 Data Privacy (11) → SMI 📁 Software Quality (11) 📁 QoS Requirements (9) 📁 Aspects (8) 📁 Cloud Computing Systems (8) 📁 High (8) 📁 Security and Privacy Issues (8) 📁 Network (7) → SMI </div> <div style="width: 50%;"> <ul style="list-style-type: none"> 📁 Distributed Environments (5) 📁 Survey (5) 📁 Distributed Systems (4) 📁 Non-functional Requirements (3) 📁 Adaptive Monitoring (2) 📁 Dependability (2) 📁 Hybrid (2) 📁 Maintaining Data (2) 📁 Maintenance (2) 📁 Nonfunctional (2) 📁 Software Quality Assurance (2) 📁 Systematic Literature Review (2) 📁 Trust is the most Prominent Problem (2) 📁 Workflow Scheduling (2) 📁 Other Topics (15) </div> </div>

Clúster	Clúster nivel 2 - "Evaluation Model (20)"
Consulta	security privacy usability metric quality interoperability usability portability attribute performance accountability assurance agility cost reliability flexibility service
Subnodos	<ul style="list-style-type: none"> 📁 Evaluation Model (20) <ul style="list-style-type: none"> 📁 Evaluate what Quality Attribute (6) 📁 Cloud Providers (5) 📁 QoS Evaluation (5) 📁 Metric Model (4) 📁 Security and Reliability Function (4) 📁 Trusted Cloud Service (4) 📁 Ability (2) 📁 Availability (2) 📁 Efficiency (2) 📁 Elasticity (2) 📁 Security and Privacy (2) 📁 Service Level (2) 📁 Software Quality (2) 📁 Other Topics (1)

Artículos

Nro.	Nombre de artículo	Referencia
1.	An Approach to IdM with Privacy in the Cloud.	(Werner et al., 2015)
2.	Quality Model for Evaluating Platform as a Service in Cloud Computing.	(Olokunde et al., 2017)
3.	A Quality Model for Evaluating Software-as-a-Service in Cloud Computing.	(J. Y. Lee et al., 2009)
4.	Towards Service Evaluation and Ranking Model for Cloud Infrastructure Selection.	(M. Z. Khan & Qamar, 2015)
5.	Quality evaluation and best service choice for cloud computing based on user preference and weights of attributes using the analytic network Process.	(Choi & Jeong, 2014)
6.	Cloud service evaluation method-based Multi-Criteria Decision-Making: A systematic literature review.	(Alabool et al., 2018)
7.	Designing an efficient methodology based on Entropy-TOPSIS for evaluating efficiency of cloud services	(R. R. Kumar & Kumar, 2017)
8.	The evaluation system for cloud service quality based on SERVQUAL	(Hu & Zhang, 2013)
9.	QoS evaluation for web services in cloud computing	(Nadanam & Rajmohan, 2012)
10.	Software metric evaluation on cloud based applications	(Phani Sheetal & Ravindranath, 2018)
11.	Research on the measurement and evaluation of trusted cloud service	(Ma et al., 2018)
12.	Modeling Correlation between QoS Attributes for Trust Computation in Cloud Computing Environments	(Mrabet et al., 2017b)
13.	A QoS Evaluation Model for Test-Bed in the Cloud Computing Environment	(Song et al., 2012)
14.	An Evaluation Model for Selecting Cloud Services from Commercially Available Cloud Providers	(Wagle et al., 2016)
15.	A Stochastic Model to Investigate Data Center Performance and QoS in IaaS Cloud Computing Systems	(Bruneo, 2014)
16.	Semantically Equivalent Model for Quality Evaluation	(Kara et al., 2017)
17.	Optimal Negotiation of Service Level Agreements for Cloud-Based Services through Autonomous Agents	(Yaqub et al., 2014)
18.	An empirical study for evaluating the performance of multi-cloud APIs	(Ré et al., 2018)
19.	A novel trust model based on SLA and behavior evaluation for	(Z. Tan et al., 2016)

Relación entre subnodos

Nro.	Nombre de artículo	Referencia																
			Evaluate what Quality Attrib & e (6)	Cloud Providers (5)	QoS Evaluation (5)	Metric Model (4)	Security and Reliability Function (4)	Trusted Cloud Service (4)	Ability (2)	Availability (2)	Efficiency (2)	Elasticity (2)	Security and Privacy (2)	Service Level (2)	Software Quality (2)	Other Topics (1)		
1	An Approach to IdM with Privacy in the Cloud.	(Werner et al., 2015)		x											x			
2	Quality Model for Evaluating Platform as a Service in Cloud Computing.	(Olokunde et al., 2012)																x
3	A Quality Model for Evaluating Software-as-a-Service in Cloud Computing.	(Lee et al., 2009)	x			x	x											
4	Towards Service Evaluation and Ranking Model for Cloud Infrastructure Selection.	(Khan & Qamar, 2016)		x														x
5	Quality evaluation and best service choice for cloud computing based on user preference and weights of attributes using the analytic network Process.	(Choi & Jeong, 2014)						x							x			
6	Cloud service evaluation method-based Multi-Criteria Decision-Making: A systematic literature review.	(Alabool et al., 2018)	x		x	x			x									
7	Designing an efficient methodology based on Entropy-TOPSIS for evaluating efficiency of cloud services	(R. R. Kumar & Kumar, 2017)													x			
8	The evaluation system for cloud service quality based on SERVQUAL	(Hu & Zhang, 2013)																x
9	QoS evaluation for web services in cloud computing	(Nadanam & Rajmohan, 2012)							x									
10	Software metric evaluation on cloud based applications	(Phani Sheeta & Ravindranath, 2018)																x
11	Research on the measurement and evaluation of trusted cloud service	(Ma et al., 2018)	x															
12	Modeling Correlation between QoS Attributes for Trust Computation in Cloud Computing Environments	(Mrabet et al., 2017)							x									
13	A QoS Evaluation Model for Test-Bed in the Cloud Computing Environment	(Song et al., 2012)																
14	An Evaluation Model for Selecting Cloud Services from Commercially Available Cloud Providers	(Wagle et al., 2016)		x														
15	A Stochastic Model to Investigate Data Center Performance and QoS in IaaS Cloud Computing Systems	(Bruneo, 2014)																
16	Semantically Equivalent Model for Quality Evaluation	(Kara et al., 2017)	x															
17	Optimal Negotiation of Service Level Agreements for Cloud-Based Services through Autonomous Agents	(Yaquib et al., 2014)		x	x													
18	An empirical study for evaluating the performance of multi-cloud APIs	(Ré et al., 2018)	x															
19	A novel trust model based on SLA and behavior evaluation for clouds	(Tan et al., 2016)		x														
20	Elasticity and scalability centric quality model for the cloud	(Abdeladim et al., 2014)	x		x													

Clúster	Clúster nivel 2 - “Reliability (19)”
Consulta	security reliability usability performance efficiency compatibility usability reliability security maintainability responsiveness tangibility empathy assurance functional availability elasticity Flexibility Finance cost scalability performance
Subnodos	<ul style="list-style-type: none"> 📁 Reliability (19) <ul style="list-style-type: none"> 📁 Quality of Service (7) 📁 Responsiveness and Empathy (5) 📁 Cloud Computing (4) 📁 Quality Model (3) 📁 Requirements (2) 📁 Systems (2) 📁 Trustworthy in Order to make a Decision (2) 📁 Types (2) 📁 Understanding (2) 📁 Other Topics (4)

Artículos

Nro.	Nombre de artículo	Referencia
1.	Quality Model of Cloud Service	(Zhou et al., 2015)
2.	Quality of service approaches in IoT: A systematic mapping	(White et al., 2017)
3.	Quality Model of Maintenance Service for Cloud Computing	(Z. Wang et al., 2015)
4.	A novel \TOPSIS\ evaluation scheme for cloud service trustworthiness combining objective and subjective aspects	(L. Lu & Yuan, 2018)
5.	A framework for selection of SAAS by evaluating the quality of Freemium model	(M. Naseer & Nazar, 2017)
6.	CLOUDQUAL A Quality Model for Cloud Services	(X. Zheng et al., 2014)
7.	The evaluation system for cloud service quality based on SERVQUAL	(Hu & Zhang, 2013)
8.	Fuzzy logic-based cloud service trustworthiness model	(Pandey & Daniel, 2016)
9.	A hybrid QoS evaluation tool based on the cloud computing platform	(Shu et al., 2015)
10.	Towards understanding uncertainty in cloud computing with risks of confidentiality, integrity, and availability	(Tchernykh et al., 2016)
11.	A novel TOPSIS evaluation scheme for cloud service trustworthiness combining objective and subjective aspects	(L. Lu & Yuan, 2018)
12.	Cloud Computing: Service Provisioning and User Requirements	(Khaddaj, 2012)
13.	Service replication taxonomy in distributed environments	(Mohamed, 2016)
14.	An Evaluation Model for Selecting Cloud Services from Commercially Available Cloud Providers	(Wagle et al., 2016)
15.	A survey on reliability in distributed systems	(W. Ahmed & Wu, 2013)
16.	Reliability evaluation suite for cloud services	(Oh et al., 2011)
17.	An analytical model to evaluate reliability of cloud computing systems in the presence of QoS requirements	(Faragardi et al., 2013)
18.	A trustworthy management approach for cloud services QoS data	(Tao et al., 2010)
19.	A trust model for resource selection in cloud computing environment	(Gholami & Arani, 2015)

Relación entre subnodos

Nro.	Nombre de artículo	Referencia																			
			Quality of Service (7)	Responsiveness and Empathy (5)	Cloud computing (4)	Quality model (3)	Requirements (2)	Systems (2)	Trustworthy in order to make a decision (2)	Types (2)	Understanding (2)	Other Topics (4)									
1	Quality Model of Cloud Service	(Zhou et al., 2015)	x	x		x															
2	Quality of service approaches in IoT: A systematic mapping	(White et al., 2017)	x			x							x								
3	Quality Model of Maintenance Service for Cloud Computing	(Z. Wang et al., 2015)	x	x	x	x															
4	A novel \TOPSIS\ evaluation scheme for cloud service trustworthiness combining objective and subjective aspects	(Lu & Yuan, 2018)	x									x									
5	A framework for selection of SAAS by evaluating the quality of Freemium model	(M. Naseer & Nazar, 2017)	x	x																x	
6	CLOUDQUAL A Quality Model for Cloud Services	(X. Zheng et al., 2014)	x	x																	
7	The evaluation system for cloud service quality based on SERVQUAL	(Hu & Zhang, 2013)	x	x																	
8	Fuzzy logic-based cloud service trustworthiness model	(Pandey & Daniel, 2016)																		x	
9	A hybrid QoS evaluation tool based on the cloud computing platform	(Shu et al., 2015)																		x	
10	Towards understanding uncertainty in cloud computing with risks of confidentiality, integrity, and availability	(Tchernykh et al., 2016)	x		x									x						x	
11	A novel TOPSIS evaluation scheme for cloud service trustworthiness combining objective and subjective aspects	(Lu & Yuan, 2018)											x								
12	Cloud Computing: Service Provisioning and User Requirements	(Khaddaj, 2012)				x		x													
13	Service replication taxonomy in distributed environments	(Mohamed, 2016)							x												
14	An Evaluation Model for Selecting Cloud Services from Commercially Available Cloud Providers	(Wagle et al., 2016)																			x
15	A survey on reliability in distributed systems	(W. Ahmed & Wu, 2013)										x									
16	Reliability evaluation suite for cloud services	(Oh et al., 2011)				x															
17	An analytical model to evaluate reliability of cloud computing systems in the presence of QoS requirements	(Faragardi et al., 2013)	x																		x
18	A trustworthy management approach for cloud services QoS data	(Tao et al., 2010)																			x
19	A trust model for resource selection in cloud computing environment	(Gholami & Arani, 2015)																			

Clúster	Clúster nivel 2 - “Availability (17)”
Consulta	Security Privacy Confidentiality Integrity Availability Accountability Usability Reliability Tangibility responsiveness elasticity performance scalability Cost reputation
Subnodos	<ul style="list-style-type: none"> 📁 Availability (17) <ul style="list-style-type: none"> 📁 Cloud (7) 📁 Quality of Service (5) 📁 Model (4) 📁 Cloud Computing (3) 📁 Provider (3) 📁 Requirements (3) 📁 Security and Privacy (3) 📁 Selection (3) 📁 Defined (2) 📁 High Performance (2) 📁 Measurement (2) 📁 Taxonomy (2) 📁 Other Topics (1)

Artículos

Nro.	Nombre de artículo	Referencia
1.	A Runtime Quality Measurement Framework for Cloud Database Service Systems	(Klems et al., 2012)
2.	Security and Privacy in Cloud Computing	(Xiao & Xiao, 2013)
3.	MCSQAM Service Quality Assessment Model in Mobile Cloud Services Environment	(Shin & Huh, 2016)
4.	A framework for selection of SAAS by evaluating the quality of Freemium model	(M. Naseer & Nazar, 2017)
5.	CLOUDQUAL A Quality Model for Cloud Services	(X. Zheng et al., 2014)
6.	A Cloud Trust Evaluation System Using Hierarchical Fuzzy Inference System for Service Selection	(Qu & Buyya, 2014)
7.	Towards understanding uncertainty in cloud computing with risks of confidentiality, integrity, and availability	(Tchernykh et al., 2016)
8.	An Approach to Identify and Monitor SLA Parameters for Storage-as-a-Service Cloud Delivery Model	(Ghosh & Ghosh, 2012)
9.	Cloud Computing: Service Provisioning and User Requirements	(Khaddaj, 2012)
10.	Service replication taxonomy in distributed environments	(Mohamed, 2016)
11.	An Evaluation Model for Selecting Cloud Services from Commercially Available Cloud Providers	(Wagle et al., 2016)
12.	Adaptive resource management for balancing availability and performance in cloud computing	(Jhavar & Piuri, 2013)
13.	Availability in the cloud_ State of the art	(Nabi et al., 2016)
14.	Quality of service for database in the cloud	(Sousa et al., 2012)
15.	A trustworthy management approach for cloud services QoS data	(Tao et al., 2010)
16.	QoS-Aware Deployment of IoT Applications Through the Fog	(Isa et al., 2016)
17.	A trust model for resource selection in cloud computing environment	(Gholami & Arani, 2015)

Relación entre subnodos

Nro.	Nombre de artículo	Referencia															
			Cloud (7)	Quality of Service (5)	Model (4)	Cloud Computing (3)	Provider (3)	Requirements (3)	Security and Privacy (3)	Selection (3)	Defined (2)	High Performance (2)	Measurement (2)	Taxonomy (2)	Other Topics (2)		
1	A Runtime Quality Measurement Framework for Cloud Database Service Systems	(Klems et al., 2012)	x	x			x						x	x			
2	Security and Privacy in Cloud Computing	(Xiao & Xiao, 2013)							x								
3	MCSQAM Service Quality Assessment Model in Mobile Cloud Services Environment	(Shin & Huh, 2016)		x													
4	A framework for selection of SAAS by evaluating the quality of Freemium model	(M. Naseer & Nazar, 2017)			x								x				
5	CLOUDQUAL A Quality Model for Cloud Services	(X. Zheng et al., 2014)															x
6	A Cloud Trust Evaluation System Using Hierarchical Fuzzy Inference System for Service Selection	(Qu & Buyya, 2014)	x					x	x	x							
7	Towards understanding uncertainty in cloud computing with risks of confidentiality, integrity, and availability	(Tchernykh et al., 2016)	x	x		x				x							
8	An Approach to Identify and Monitor SLA Parameters for Storage-as-a-Service Cloud Delivery Model	(Ghosh & Ghosh, 2012)	x		x		x										
9	Cloud Computing: Service Provisioning and User Requirements	(Khaddaj, 2012)	x			x		x									
10	Service replication taxonomy in distributed environments	(Mohamed, 2016)							x								x
11	An Evaluation Model for Selecting Cloud Services from Commercially Available Cloud Providers	(Wagle et al., 2016)	x		x		x				x			x			
12	Adaptive resource management for balancing availability and performance in cloud computing	(Jhavar & Piuri, 2013)	x			x											
13	Availability in the cloud_ State of the art	(Nabi, Toeroe, & Khendek, 2016)									x						x
14	Quality of service for database in the cloud	(Sousa, Moreira, Santos, & Machado, 2012)		x									x				
15	A trustworthy management approach for cloud services QoS data	(Tao, Chang, Yi, & Gu, 2010b)											x				
16	QoS-Aware Deployment of IoT Applications Through the Fog	(Isa, Brogi, & Forti, 2016)			x												
17	A trust model for resource selection in cloud computing environment	(Gholami & Arani, 2015)		x													

Clúster	Clúster nivel 2 - “Service Provider (17)”
Consulta	Security privacy Performance availability scalability profitability Price usability reputation stability cost usability reliability efficiency
Subnodos	<ul style="list-style-type: none"> 📁 Service Provider (17) <ul style="list-style-type: none"> 📁 Cloud Monitoring (5) 📁 Integrity (3) 📁 Parameters Availability (3) 📁 Service Selection (3) 📁 Attributes (2) 📁 Guarantees (2) 📁 High (2) 📁 Measurement (2) 📁 Optimal (2) 📁 Trustworthy in Order to make a Decision (2) 📁 Web Service (2) 📁 Other Topics (2)

Artículos

Nro.	Nombre de artículo	Referencia
1.	A broker-based framework for standardization and management of Cloud Security-SLAs	(Halabi & Bellaiche, 2018)
2.	A Runtime Quality Measurement Framework for Cloud Database Service Systems	(Klems et al., 2012)
3.	Integrated QoS Utility-Based Model for Cloud Computing Service Provider Selection	(Salama et al., 2012)
4.	An End-to-End QoS Mapping Approach for Cloud Service Selection	(Karim et al., 2013)
5.	A QoS evaluation model in the environment of cloud computing	(Z. E. Wang & Liu, 2012)
6.	A novel \TOPSIS\ evaluation scheme for cloud service trustworthiness combining objective and subjective aspects	(L. Lu & Yuan, 2018)
7.	An Approach to Identify and Monitor SLA Parameters for Storage-as-a-Service Cloud Delivery Model	(Ghosh & Ghosh, 2012)
8.	Toward cloud computing QoS architecture analysis of cloud systems and cloud services	(Ghahramani et al., 2017)
9.	Communication of Technical QoS among Cloud Brokers	(Lim & Thiran, 2014)
10.	An Evaluation Model for Selecting Cloud Services from Commercially Available Cloud Providers	(Wagle et al., 2016)
11.	MCTModel A multi-clouds trust model based on SLA in cloud computing	(Z.-H. Tan et al., 2017)
12.	A framework for IaaS-to-SaaS monitoring of BPEL processes in the Cloud Design and evaluation	(Grati et al., 2014)
13.	Optimal Negotiation of Service Level Agreements for Cloud-Based Services through Autonomous Agents	(Yaqub et al., 2014)
14.	A novel TOPSIS evaluation scheme for cloud service trustworthiness combining objective and subjective aspects	(L. Lu & Yuan, 2018)
15.	A novel trust model based on SLA and behavior evaluation for clouds	(Z. Tan et al., 2016)
16.	The interplay between timeliness and scalability in cloud monitoring systems	(Da Cunha Rodrigues et al., 2016)
17.	End-to-End QoS on Shared Clouds for Highly Dynamic, Large-Scale Sensing Data Streams	(Tolosana-Calasanz et al., 2012)

Relación entre subnodos

Nro.	Nombre de artículo	Referencia															
			Cloud monitoring (5)	Integrity (5)	Parameters Availability (3)	Service Selection (3)	Attrib Uses (2)	Guarantees (2)	Measurement (2)	Optimal (2)	Trustworthy in order to make a decision (2)	Web Service (2)	Other topics (1)				
1	A broker-based framework for standardization and management of Cloud Security-SLAs	(Halabi & Bellaiche, 2018)		x					x								
2	A Runtime Quality Measurement Framework for Cloud Database Service Systems	(Klems et al., 2012)															x
3	Integrated QoS Utility-Based Model for Cloud Computing Service Provider Selection	(Salama et al., 2012)		x													x
4	An End-to-End QoS Mapping Approach for Cloud Service Selection	(Karim et al., 2013)															x
5	A QoS evaluation model in the environment of cloud computing	(Z. E. Wang & Liu, 2012)															x
6	A novel (TOPSIS) evaluation scheme for cloud service trustworthiness combining objective and subjective aspects	(Lu & Yuan, 2018)	x														x
7	An Approach to Identify and Monitor SLA Parameters for Storage-as-a-Service Cloud Delivery Model	(Ghosh & Ghosh, 2012)	x														x
8	Toward cloud computing QoS architecture analysis of cloud systems and cloud services	(Ghahramani et al., 2017)															x
9	Communication of Technical QoS among Cloud Brokers	(Lim & Thiran, 2014)															x
10	An Evaluation Model for Selecting Cloud Services from Commercially Available Cloud Providers	(Wagle et al., 2016)															x
11	MCTModel A multi-clouds trust model based on SLA in cloud computing	(Z.-H. Tan et al., 2017)		x													x
12	A framework for IaaS-to-SaaS monitoring of BPEL processes in the Cloud Design and evaluation	(Grati et al., 2014)	x														x
13	Optimal Negotiation of Service Level Agreements for Cloud-Based Services through Autonomous Agents	(Yaqub et al., 2014)															x
14	A novel TOPSIS evaluation scheme for cloud service trustworthiness combining objective and subjective aspects	(Lu & Yuan, 2018)	x														x
15	A novel trust model based on SLA and behavior evaluation for clouds	(Z. Tan et al., 2016)		x													x
16	The interplay between timeliness and scalability in cloud monitoring systems	(Da Cunha Rodrigues et al., 2016)	x														
17	End-to-End QoS on Shared Clouds for Highly Dynamic, Large-Scale Sensing Data Streams	(Tolosana-Calasanz et al., 2012)															

Clúster	Clúster nivel 2 - “Service Selection (16)”
Consulta	Services Accountability attributes security privacy agility assurance financial performance usability Performance cost profitability Price Reputation Availability Reliability
Subnodos	<ul style="list-style-type: none"> 📁 Service Selection (16) <ul style="list-style-type: none"> 📁 Cloud Providers (7) 📁 Trust (3) 📁 Web Services (3) 📁 Designed (2) 📁 Efficient (2) 📁 Fuzzy (2) 📁 Improved (2) 📁 Integrity (2) 📁 QoS Properties (2) 📁 Rank (2) 📁 Set (2)

Artículos

Nro.	Nombre de artículo	Referencia
1.	Bootstrapping quality of Web Services	(Aljazzaf, 2015)
2.	Towards Service Evaluation and Ranking Model for Cloud Infrastructure Selection	(M. Z. Khan & Qamar, 2015)
3.	A cloud service selection model using improved ranked voting method	(Baranwal & Vidyarthi, 2016)
4.	Designing an efficient methodology based on Entropy-TOPSIS for evaluating efficiency of cloud services	(R. R. Kumar & Kumar, 2017)
5.	Integrated QoS Utility-Based Model for Cloud Computing Service Provider Selection	(Salama et al., 2012)
6.	An End-to-End QoS Mapping Approach for Cloud Service Selection	(Karim et al., 2013)
7.	Fuzzy Cloud Service Selection Framework	(Tajvidi et al., 2014)
8.	A Cloud Trust Evaluation System Using Hierarchical Fuzzy Inference System for Service Selection	(Qu & Buyya, 2014)
9.	Modeling Correlation between QoS Attributes for Trust Computation in Cloud Computing Environments	(Mrabet et al., 2017b)
10.	An Approach to Identify and Monitor SLA Parameters for Storage-as-a-Service Cloud Delivery Model	(Ghosh & Ghosh, 2012)
11.	Communication of Technical QoS among Cloud Brokers	(Lim & Thiran, 2014)
12.	An Evaluation Model for Selecting Cloud Services from Commercially Available Cloud Providers	(Wagle et al., 2016)
13.	CloudRec: A Mobile Cloud Service Recommender System Based on Adaptive QoS Management	(Tang & Yan, 2015)
14.	An Efficient Algorithm for Partially Matched Web Services Based on Consumer's QoS Requirements	(M. Ahmed et al., 2014)
15.	A Novel QoS Prediction Approach for Cloud Service Based on Bayesian Networks Model	(Zhang et al., 2016)
16.	A trustworthy management approach for cloud services QoS data	(Tao et al., 2010)

Relación entre subnodos

Nro.	Nombre de artículo	Referencia															
			Cloud Providers (7)	Trust (3)	Web Services (3)	Designed (2)	Efficient (2)	Fuzzy (2)	Improved (2)	Integrity (2)	QoS Properties (2)	Rank (2)	S&T (2)				
1	Bootstrapping quality of Web Services	(Aljazzaf, 2015)			x											x	
2	Towards Service Evaluation and Ranking Model for Cloud Infrastructure Selection	(Khan & Qamar, 2016)														x	
3	A cloud service selection model using improved ranked voting method	(Baranwal & Vidyarthi, 2016)	x						x							x	
4	Designing an efficient methodology based on Entropy-TOPSIS for evaluating efficiency of cloud services	(R. R. Kumar & Kumar, 2017)				x	x										
5	Integrated QoS Utility-Based Model for Cloud Computing Service Provider Selection	(Salama et al., 2012)	x			x				x							
6	An End-to-End QoS Mapping Approach for Cloud Service Selection	(Karim et al., 2013)		x	x												
7	Fuzzy Cloud Service Selection Framework	(Tajvidi, Ranjan, Kolodziej, & Wang, 2014)	x						x								
8	A Cloud Trust Evaluation System Using Hierarchical Fuzzy Inference System for Service Selection	(Qu & Buyya, 2014)		x					x	x							
9	Modeling Correlation between QoS Attributes for Trust Computation in Cloud Computing Environments	(Mrabet et al., 2017)		x													
10	An Approach to Identify and Monitor SLA Parameters for Storage-as-a-Service Cloud Delivery Model	(Ghosh & Ghosh, 2012)	x														
11	Communication of Technical QoS among Cloud Brokers	(Lim & Thiran, 2014)	x												x		
12	An Evaluation Model for Selecting Cloud Services from Commercially Available Cloud Providers	(Wagle et al., 2016)	x														
13	CloudRec: A Mobile Cloud Service Recommender System Based on Adaptive QoS Management	(Tang & Yan, 2015)								x							
14	An Efficient Algorithm for Partially Matched Web Services Based on Consumer's QoS Requirements	(M. Ahmed, Liu, Hardy, & Yuan, 2014)	x		x		x										
15	A Novel QoS Prediction Approach for Cloud Service Based on Bayesian Networks Model	(Zhang, Han, Li, Leung, & Song, 2016)													x		
16	A trustworthy management approach for cloud services QoS data	(Tao et al., 2010b)															x

Clúster	Clúster nivel 2 - “Quality Attributes (15)”
Consulta	Robustness Service Attribute Accountability Agility Assurance Financial Performance Security Privacy Usability Reliability Tangibility profitability transparency security privacy interoperability availability scalability Cost
Subnodos	<ul style="list-style-type: none"> 📁 Quality Attributes (15) <ul style="list-style-type: none"> 📁 Service Providers (7) 📁 Quality Model (5) 📁 Service Evaluation (5) 📁 Cloud Computing (4) 📁 QoS Performance (3) 📁 Selection (3) 📁 Assessment (2) 📁 Important (2) 📁 Review (2) 📁 Service Consumers (2) 📁 Web Service (2)

Artículos

Nro.	Nombre de artículo	Referencia
1.	A Quality Model for Evaluating Software-as-a-Service in Cloud Computing	(J. Y. Lee et al., 2009)
2.	A framework for selection of best cloud service provider using ranked voting method	(Baranwal & Vidyarthi, 2014)
3.	Towards Service Evaluation and Ranking Model for Cloud Infrastructure Selection	(M. Z. Khan & Qamar, 2015)
4.	Cloud service evaluation method-based Multi-Criteria Decision-Making: A systematic literature review	(Alabool et al., 2018)
5.	A framework for selection of SAAS by evaluating the quality of Freemium model	(M. Naseer & Nazar, 2017)
6.	Integrated QoS Utility-Based Model for Cloud Computing Service Provider Selection	(Salama et al., 2012)
7.	Towards an Approach for Trustworthiness Assessment of Software as a Service	(Medeiros et al., 2017)
8.	Trust management in cloud computing A survey	(Monir et al., 2015)
9.	Research on the measurement and evaluation of trusted cloud service	(Ma et al., 2018)
10.	An Overview of Software Functionality Service A Systematic Literature Review	(Salleh et al., 2017)
11.	Service replication taxonomy in distributed environments	(Mohamed, 2016)
12.	An Evaluation Model for Selecting Cloud Services from Commercially Available Cloud Providers	(Wagle et al., 2016)
13.	Reliability evaluation suite for cloud services	(Oh et al., 2011)
14.	An empirical study for evaluating the performance of multi-cloud APIs	(Ré et al., 2018)vv
15.	A trustworthy QoS-based collaborative filtering approach for web service discovery	(Lin et al., 2014)

Relación entre subnodos

Nro.	Nombre de artículo	Referencia												
			Service Providers (7)	Quality Model (5)	Service Evaluation (5)	Cloud Computing (5)	QoS Performance (4)	Selection (3)	Assessment (2)	Important (2)	Review (2)	Service Consumer (2)	Web Service (2)	
1	A Quality Model for Evaluating Software-as-a-Service in Cloud Computing	(Lee et al., 2009)	x	x	x	x					x		x	
2	A framework for selection of best cloud service provider using ranked voting method	(Baranwal & Vidyarthi, 2014)	x											
3	Towards Service Evaluation and Ranking Model for Cloud Infrastructure Selection	(Khan & Qamar, 2016)	x		x			x						
4	Cloud service evaluation method-based Multi-Criteria Decision-Making: A systematic literature review	(Alabool et al., 2018)			x							x		
5	A framework for selection of SAAS by evaluating the quality of Freemium model	(M. Naseer & Nazar, 2017)		x										
6	Integrated QoS Utility-Based Model for Cloud Computing Service Provider Selection	(Salama et al., 2012)	x	x		x	x	x					x	
7	Towards an Approach for Trustworthiness Assessment of Software as a Service	(Medeiros, Ivaki, Costa, & Vieira, 2017)							x	x				
8	Trust management in cloud computing A survey	(Challagidad & Birje, 2018)	x						x					
9	Research on the measurement and evaluation of trusted cloud service	(Ma et al., 2018)	x	x	x									
10	An Overview of Software Functionality Service A Systematic Literature Review	(Salleh, Bahari, & Zakaria, 2017)		x		x						x		
11	Service replication taxonomy in distributed environments	(Mohamed, 2016)					x							
12	An Evaluation Model for Selecting Cloud Services from Commercially Available Cloud Providers	(Wagle et al., 2016)	x		x			x						
13	Reliability evaluation suite for cloud services	(Oh et al., 2011)												x
14	An empirical study for evaluating the performance of multi-cloud APIs	(Ré et al., 2018)w				x								
15	A trustworthy QoS-based collaborative filtering approach for web service discovery	(Lin, Lai, Wu, & Lo, 2014)					x							x

Clúster	Clúster nivel 2 - “Measurement (12)”
Consulta	security privacy issues performance Usability service availability elastic scalability Accountability Agility Assurance Financial Quality Metrics Attributes Reliability Cost
Subnodos	<ul style="list-style-type: none"> 📁 Measurement (12) <ul style="list-style-type: none"> 📁 Quality of Service QoS (5) 📁 Service Provider (4) 📁 Assess (3) 📁 Paper (3) 📁 Security and Privacy (3) 📁 Cloud Computing (2) 📁 Framework (2) 📁 High (2) 📁 Model (2) 📁 QoS Metrics (2) 📁 Other Topics (1)

Artículos

Nro.	Nombre de artículo	Referencia
1.	An Approach to IdM with Privacy in the Cloud	(Werner et al., 2015)
2.	Q-aware: Quality-of-service-based cloud resource provisioning	(S. Singh & Chana, 2015)v
3.	A Runtime Quality Measurement Framework for Cloud Database Service Systems	(Klems et al., 2012)
4.	A framework for selection of best cloud service provider using ranked voting method	(Baranwal & Vidyarthi, 2014)
5.	Trust management in cloud computing A survey	(Monir et al., 2015)
6.	Research on the measurement and evaluation of trusted cloud service	(Ma et al., 2018)
7.	From QoD to QoS Data quality issues in cloud computing	(Pawluk et al., 2011)
8.	Scalability Analysis of Cloud Software Services	(Brataas et al., 2017)
9.	An Evaluation Model for Selecting Cloud Services from Commercially Available Cloud Providers	(Wagle et al., 2016)
10.	Efficient QOS with multiple constraints for epigenomic scientific workflow model in SAAS cloud	(Sivaraj & Muniraj, 2014)
11.	Assessing Measurements of QoS for Global Cloud Computing Services	(Pedersen et al., 2011)
12.	Measuring the latency of cloud gaming systems	(Chen et al., 2011)

Relación entre subnodos

Nro.	Nombre de artículo	Referencia															
			Quality of Service QoS (5)	Service Provider (4)	Assess (3)	Security and Privacy (3)	Approach (2)	Cloud Computing (2)	Framework (2)	High (2)	Model (2)	QoS Metric (2)					
1	An Approach to IdM with Privacy in the Cloud	(Werner et al., 2015)		x			x	x									
2	Q-aware: Quality of service based cloud resource provisioning	(S. Singh & Chana, 2015)	x							x							x
3	A Runtime Quality Measurement Framework for Cloud Database Service Systems	(Klems et al., 2012)		x						x	x						
4	A framework for selection of best cloud service provider using ranked voting method	(Baranwal & Vidyarathi, 2014)	x				x										x
5	Trust management in cloud computing A survey	(Challagidad & Birje, 2018)				x	x										
6	Research on the measurement and evaluation of trusted cloud service	(Ma et al., 2018)															x
7	From QoD to QoS Data quality issues in cloud computing	(Pawluk, Litoiu, & Cercone, 2011)	x	x					x			x					
8	Scalability Analysis of Cloud Software Services	(Brataas, Herbst, Ivansek, & Polutnik, 2017)							x								
9	An Evaluation Model for Selecting Cloud Services from Commercially Available Cloud Providers	(Wagle et al., 2016)		x													x
10	Efficient QoS with multiple constraints for epigenomic scientific workflow model in SAAS cloud	(Sivaraj & Muniraj, 2014)	x														
11	Assessing Measurements of QoS for Global Cloud Computing Services	(Pedersen et al., 2011)	x			x				x							
12	Measuring the latency of cloud gaming systems	(Chen, Chang, Tseng, Huang, & Lei, 2011)				x											

Clúster	Clúster nivel 2 - “Metric Model (12)”
Consulta	Usability interoperability usability portability Performance Efficiency Compatibility Metric Reliability Maintainability Robustness accountability agility assurance financial security privacy
Subnodos	<ul style="list-style-type: none"> 📁 Metric Model (12) <ul style="list-style-type: none"> 📁 Cloud Computing (6) 📁 Quality Model for Evaluating (4) 📁 Attribute (2) 📁 Cloud Computing Systems (2) 📁 Composition (2) 📁 Key (2) 📁 Method (2) 📁 Support (2) 📁 Systematic (2) 📁 Value (2)

Artículos

Nro.	Nombre de artículo	Referencia
1.	Quality model for evaluating platform as a service in cloud computing	(Olokunde et al., 2017)
2.	Quality of service approaches in IoT A systematic mapping	(White et al., 2017)
3.	A Quality Model for Evaluating Software-as-a-Service in Cloud Computing	(J. Y. Lee et al., 2009)
4.	QoE metrics aggregation for hierarchical Service Level Agreement in Cross-Layered SLA architecture	(Shin & Huh, 2015)
5.	A cloud service selection model using improved ranked voting method	(Baranwal & Vidyarthi, 2016)
6.	Cloud service evaluation method-based Multi-Criteria Decision-Making A systematic literature review	(Alabool et al., 2018)
7.	Software metric evaluation on cloud-based applications	(Phani Sheetal & Ravindranath, 2018)
8.	Reliability evaluation suite for cloud services	(Oh et al., 2011)
9.	A Stochastic Model to Investigate Data Center Performance and QoS in IaaS Cloud Computing Systems	(Bruneo, 2014)
10.	SLA-Based Service Composition Model with Semantic Support	(H. Liu et al., 2012)
11.	Semantically Equivalent Model for Quality Evaluation	(Kara et al., 2017)
12.	An analytical model to evaluate reliability of cloud computing systems in the presence of QoS requirements	(Faragardi et al., 2013)

Relación entre subnodos

Nro.	Nombre de artículo	Referencia													
			Cloud Computing (6)	Quality Model for Evaluating (4)	Attribute (2)	Cloud Computing System (2)	Composition (2)	Key (2)	Method (2)	Support (2)	Systematic (2)	Value (2)			
1	Quality model for evaluating platform as a service in cloud computing	(Olokunde et al., 2017)	x	x											x
2	Quality of service approaches in IoT A systematic mapping	(White et al., 2017)													x
3	A Quality Model for Evaluating Software-as-a-Service in Cloud Computing	(Lee et al., 2009)	x	x	x										
4	QoE metrics aggregation for hierarchical Service Level Agreement in Cross-Layered SLA architecture	(Shin & Huh, 2015)									x				
5	A cloud service selection model using improved ranked voting method	(Baranwal & Vidyarthi, 2016)	x					x	x						
6	Cloud service evaluation method-based Multi-Criteria Decision-Making A systematic literature review	(Alabool et al., 2018)													x
7	Software metric evaluation on cloud based applications	(Phani Sheetal & Ravindranath, 2018)		x						x					
8	Reliability evaluation suite for cloud services	(Oh et al., 2011)	x		x										
9	A Stochastic Model to Investigate Data Center Performance and QoS in IaaS Cloud Computing Systems	(Bruneo, 2014)	x			x									
10	SLA-Based Service Composition Model with Semantic Support	(H. Liu, Bu, & Cai, 2012)						x			x				x
11	Semantically Equivalent Model for Quality Evaluation	(Kara et al., 2017)		x				x							
12	An analytical model to evaluate reliability of cloud computing systems in the presence of QoS requirements	(Faragardi et al., 2013)	x			x			x						

Clúster	Clúster nivel 2 - “Data Privacy (11)”
Consulta	Security privacy service Usability confidentiality efficiency performance reliability Quality
Subnodos	<ul style="list-style-type: none"> 📁 Data Privacy (11) <ul style="list-style-type: none"> 📁 Cloud Computing (5) 📁 Data Confidentiality (3) 📁 Quality of Service (3) 📁 Approach (2) 📁 Issue of Service Quality (2) 📁 Level (2) 📁 QoS (2) 📁 Response Time (2) 📁 Trust (2)

Artículos

Nro.	Nombre de artículo	Referencia
1.	A broker-based framework for standardization and management of Cloud Security-SLAs	(Halabi & Bellaiche, 2018)
2.	An Approach to IdM with Privacy in the Cloud	(Werner et al., 2015)
3.	Cloud Computing: The Issue of Service Quality: An Overview of Cloud Service Level Management Architectures	(Motta et al., 2013)
4.	Survey of data and storage security in cloud computing	(Rajeswari & Kalaiselvi, 2018)
5.	Toward a framework for data quality in cloud-based health information system	(Almutiry et al., 1998)
6.	An End-to-End QoS Mapping Approach for Cloud Service Selection	(Karim et al., 2013)
7.	A hybrid QoS evaluation tool based on the cloud computing platform	(Shu et al., 2015)
8.	A survey on data confidentiality and privacy in cloud computing	(Aloraini & Hammoudeh, 2017)
9.	Trustworthy mechanisms for selecting cloud service providers	(Elizabeth et al., 2014)
10.	A Hybrid Model for Ranking Cloud Services	(De Benedetti et al., 2015)
11.	A novel trust model for selection of Cloud Service Provider	(M. K. Naseer et al., 2014)

Relación entre subnodos

Nro.	Nombre de artículo	Referencia											
			Cloud Computing (5)	Data Confidentiality (3)	Quality of Service (3)	Approach (3)	Issue of Service Quality (2)	Level (2)	QoS (2)	Response Time (2)	Trust (2)		
1	A broker-based framework for standardization and management of Cloud Security-SLAs	(Halabi & Bellaiche, 2018)						x					
2	An Approach to IdM with Privacy in the Cloud	(Werner et al., 2015)	x			x							
3	Cloud Computing: The Issue of Service Quality: An Overview of Cloud Service Level Management Architectures	(Motta, You, Sacco, & Sfondrini, 2013)	x		x		x	x	x				
4	Survey of data and storage security in cloud computing	(Rajeswari & Kalaiselvi, 2018)	x	x									
5	Toward a framework for data quality in cloud-based health information system	(Almutiry, Wills, Alwabel, Crowder, & Walters, 1998)		x									
6	An End-to-End QoS Mapping Approach for Cloud Service Selection	(Karim et al., 2013)	x			x			x	x	x		
7	A hybrid QoS evaluation tool based on the cloud computing platform	(Shu et al., 2015)											x
8	A survey on data confidentiality and privacy in cloud computing	(Aloraini & Hammoudeh, 2017)	x	x									
9	Trustworthy mechanisms for selecting cloud service providers	(Elizabeth, Ramya, Prakash, & Uthariaraj, 2014)			x		x						
10	A Hybrid Model for Ranking Cloud Services	(De Benedetti, Durso, Messina, Pappalardo, & Santoro, 2015)									x		
11	A novel trust model for selection of Cloud Service Provider	(M. K. Naseer et al., 2014)			x								

Clúster	Clúster nivel 2 - “Software Quality (11)”
Consulta	Usability Performance Security Reliability Tangibility availability security privacy interoperability transparency scalability elasticity efficiency agility cost
Subnodos	<ul style="list-style-type: none"> 📁 Software Quality (11) <ul style="list-style-type: none"> 📁 Model (6) 📁 Quality Model (4) 📁 Service (4) 📁 Evaluated (3) 📁 Assess (2) 📁 Cloud (2) 📁 Engineers (2) 📁 Important (2) 📁 Trust (2) 📁 Other Topics (1)

Artículos

Nro.	Nombre de artículo	Referencia
1.	Quality model for evaluating platform as a service in cloud computing	(Olokunde et al., 2017)
2.	Quality evaluation and best service choice for cloud computing based on user preference and weights of attributes using the analytic network process	(Choi & Jeong, 2014)
3.	A framework for selection of SAAS by evaluating the quality of Freemium model	(M. Naseer & Nazar, 2017)
4.	Towards an Approach for Trustworthiness Assessment of Software as a Service	(Medeiros et al., 2017)
5.	Software metric evaluation on cloud-based applications	(Phani Sheetal & Ravindranath, 2018)
6.	Beyond simulation Composing scalability, elasticity, and efficiency analyses from preexisting analysis results	(S. Lehrig & Becker, 2015)
7.	Quality Assessment of Software as a Service on Cloud Using Fuzzy Logic	(Baliyan & Kumar, 2013)
8.	An Overview of Software Functionality Service A Systematic Literature Review	(Salleh et al., 2017)
9.	Experience Availability Tail-Latency Oriented Availability in Software-Defined Cloud Computing	(Cai et al., 2017)
10.	Scalability Analysis of Cloud Software Services	(Brataas et al., 2017)
11.	Efficient QOS with multiple constraints for epigenomic scientific workflow model in SAAS cloud	(Sivaraj & Muniraj, 2014)

Relación entre subnodos

Nro.	Nombre de artículo	Referencia												
			Model (6)	Quality Model (4)	Service (4) (Evaluated (3)	Assess (2)	Cloud (2)	Engineers (2)	Important (2)	Trust (2)	Other Topics (1)			
1	Quality model for evaluating platform as a service in cloud computing	(Olokunde et al., 2017)	x	x	x	x		x		x				
2	Quality evaluation and best service choice for cloud computing based on user preference and weights of attributes using the analytic network process	(Choi & Jeong, 2014)												x
3	A framework for selection of SAAS by evaluating the quality of Freemium model	(M. Naseer & Nazar, 2017)	x	x										
4	Towards an Approach for Trustworthiness Assessment of Software as a Service	(Medeiros et al., 2017)			x		x			x	x			
5	Software metric evaluation on cloud based applications	(Phani Sheetal & Ravindranath, 2018)	x	x		x								
6	Beyond simulation Composing scalability, elasticity, and efficiency analyses from preexisting analysis results	(Lehrig & Becker, 2015)								x				
7	Quality Assessment of Software as a Service on Cloud Using Fuzzy Logic	(Baliyan & Kumar, 2013)	x							x				
8	An Overview of Software Functionality Service A Systematic Literature Review	(Salleh et al., 2017)	x	x	x									
9	Experience Availability Tail-Latency Oriented Availability in Software-Defined Cloud Computing	(Cai, Zhang, Zhou, Zhao, & Li, 2017)	x			x	x							
10	Scalability Analysis of Cloud Software Services	(Brataas et al., 2017)			x			x						
11	Efficient QOS with multiple constraints for epigenomic scientific workflow model in SAAS cloud	(Sivaraj & Muniraj, 2014)										x		

Clúster	Clúster nivel 2 - “QoS Requirements (9)”
Consulta	Performance profitability security privacy attribute availability scalability reliability reputation
Subnodos	<ul style="list-style-type: none"> 📁 QoS Requirements (9) <ul style="list-style-type: none"> 📁 Cloud Computing (4) 📁 QoS Requirements (4) 📁 Web Services (3) 📁 Considered (2) 📁 Environment (2) 📁 Experience (2) 📁 Table (2)

Artículos

Nro.	Nombre de artículo	Referencia
1.	Integrated QoS Utility-Based Model for Cloud Computing Service Provider Selection	(Salama et al., 2012)
2.	A Cloud Trust Evaluation System Using Hierarchical Fuzzy Inference System for Service Selection	(Qu & Buyya, 2014)
3.	Service replication taxonomy in distributed environments	(Mohamed, 2016)
4.	A framework for IaaS-to-SaaS monitoring of BPEL processes in the Cloud Design and evaluation	(Grati et al., 2014)
5.	An Efficient Algorithm for Partially Matched Web Services Based on Consumer's QoS Requirements	(M. Ahmed et al., 2014)
6.	A Stochastic Model to Investigate Data Center Performance and QoS in IaaS Cloud Computing Systems	(Bruneo, 2014)
7.	A trustworthy QoS-based collaborative filtering approach for web service discovery	(Lin et al., 2014)
8.	A trustworthy management approach for cloud services QoS data	(Tao et al., 2010)
9.	Assessing Measurements of QoS for Global Cloud Computing Services	(Pedersen et al., 2011)

Relación entre subnodos

Nro.	Nombre de artículo	Referencia							
			Cloud Computing (4)	QoS Requirements (4)	Web Services (3)	Considered (2)	Environment (2)	Experience (2)	Table (2)
1	Integrated QoS Utility-Based Model for Cloud Computing Service Provider Selection	(Salama et al., 2012)	x	x		x			
2	A Cloud Trust Evaluation System Using Hierarchical Fuzzy Inference System for Service Selection	(Qu & Buyya, 2014)		x					
3	Service replication taxonomy in distributed environments	(Mohamed, 2016)		x		x			
4	A framework for IaaS-to-SaaS monitoring of BPEL processes in the Cloud Design and evaluation	(Grati et al., 2014)	x		x	x			
5	An Efficient Algorithm for Partially Matched Web Services Based on Consumer's QoS Requirements	(M. Ahmed et al., 2014)		x	x				x
6	A Stochastic Model to Investigate Data Center Performance and QoS in IaaS Cloud Computing Systems	(Bruneo, 2014)	x						
7	A trustworthy QoS-based collaborative filtering approach for web service discovery	(Lin et al., 2014)			x				x
8	A trustworthy management approach for cloud services QoS data	(Tao et al., 2010a)				x		x	
9	Assessing Measurements of QoS for Global Cloud Computing Services	(Pedersen et al., 2011)	x					x	

Clúster	Clúster nivel 2 - “Aspects (8)”
Consulta	Security Privacy Usability profitability cost reliability availability performance
Subnodos	<ul style="list-style-type: none"> 📁 Aspects (8) 📁 Cloud Services (5) 📁 Cloud Computing (2) 📁 Management (2) 📁 Quality of Service (2) 📁 Trust has become a Necessary Security Relationship (2)

Artículos

Nro.	Nombre de artículo	Referencia
1.	Evaluating Security and Privacy in Cloud Services	(Abuhussein et al., 2016)v
2.	Quality evaluation and best service choice for cloud computing based on user preference and weights of attributes using the analytic network process	(Choi & Jeong, 2014)
3.	Integrated QoS Utility-Based Model for Cloud Computing Service Provider Selection	(Salama et al., 2012)
4.	A novel \TOPSIS\ evaluation scheme for cloud service trustworthiness combining objective and subjective aspects	(L. Lu & Yuan, 2018)
5.	Cloud Computing Service Provisioning and User Requirements	(Khaddaj, 2012)
6.	A survey on elasticity management in PaaS systems	(Muñoz-Escóí & Bernabéu-Aubán, 2017)
7.	A novel TOPSIS evaluation scheme for cloud service trustworthiness combining objective and subjective aspects	(L. Lu & Yuan, 2018)
8.	Quality of service for database in the cloud	(Sousa et al., 2012)

Relación entre subnodos

Nro.	Nombre de artículo	Referencia					
			Cloud Services (5)	Cloud Computing (2)	Management (2)	Quality of Service (2)	Trust has become a Necessary Security Relationship (2)
1	Evaluating Security and Privacy in Cloud Services	(Abuhussein, Alsubaei, Shiva, & Sheldon, 2016)v	x				
2	Quality evaluation and best service choice for cloud computing based on user preference and weights of attributes using the analytic network process	(Choi & Jeong, 2014)			x		
3	Integrated QoS Utility-Based Model for Cloud Computing Service Provider Selection	(Salama et al., 2012)	x	x		x	
4	A novel \TOPSIS\ evaluation scheme for cloud service trustworthiness combining objective and subjective aspects	(L. Lu & Yuan, 2018)	x				x
5	Cloud Computing Service Provisioning and User Requirements	(Khaddaj, 2012)	x	x			
6	A survey on elasticity management in PaaS systems	(Muñoz-Escóí & Bernabéu-Aubán, 2017)			x		
7	A novel TOPSIS evaluation scheme for cloud service trustworthiness combining objective and subjective aspects	(L. Lu & Yuan, 2018)	x				x
8	Quality of service for database in the cloud	(Sousa et al., 2012)				x	

Clúster	Clúster nivel 2 - “Cloud Computing System (8)”
Consulta	Performance Usability Metrics Security Cost Privacy Reliability
Subnodos	<ul style="list-style-type: none"> 📁 Cloud Computing Systems (8) <ul style="list-style-type: none"> 📁 Availability (2) 📁 Efficiency (2) 📁 Evaluation (2) 📁 Workflow Scheduling Approaches in Cloud Computing Environments (2) 📁 Other Topics (1)

Artículos

Nro.	Nombre de artículo	Referencia
1.	Quality evaluation and best service choice for cloud computing based on user preference and weights of attributes using the analytic network process	(Choi & Jeong, 2014)
2.	Cost-aware challenges for workflow scheduling approaches in cloud computing environments: Taxonomy and opportunities .	(Alkhanak et al., 2015)
3.	Systematically deriving quality metrics for cloud computing systems	(Becker et al., 2015)
4.	A QoS Evaluation Model for Test-Bed in the Cloud Computing Environment	(Song et al., 2012)
5.	A trust model in cloud computing based on fuzzy logic	(Ritu & Jain, 2017)
6.	A Stochastic Model to Investigate Data Center Performance and QoS in IaaS Cloud Computing Systems	(Bruneo, 2014)v
7.	Cost-aware challenges for workflow scheduling approaches in cloud computing environments: Taxonomy and opportunities	(Alkhanak et al., 2015)
8.	An analytical model to evaluate reliability of cloud computing systems in the presence of QoS requirements	(Faragardi et al., 2013)

Relación entre subnodos

Nro.	Nombre de artículo	Referencia						
			Availability (2)	Efficiency (2)	Evaluation (2)	Workflow Scheduling Approaches in Cloud Computing Environments (2)	Other Topics (2)	
1	Quality evaluation and best service choice for cloud computing based on user preference and weights of attributes using the analytic network process	(Choi & Jeong, 2014)	x	x				
2	Cost-aware challenges for workflow scheduling approaches in cloud computing environments: Taxonomy and opportunities .	(Alkhanak, Lee, & Khan, 2015)				x		
3	Systematically deriving quality metrics for cloud computing systems	(Becker, Lehrig, & Becker, 2015)		x				
4	A QoS Evaluation Model for Test-Bed in the Cloud Computing Environment	(Song et al., 2012)			x			
5	A trust model in cloud computing based on fuzzy logic	(Ritu & Jain, 2017)					x	
6	A Stochastic Model to Investigate Data Center Performance and QoS in IaaS Cloud Computing Systems	(Bruneo, 2014)v			x		x	
7	Cost-aware challenge for workflow scheduling approaches in cloud computing environments Taxonomy and opportunities	(Alkhanak et al., 2015)				x		
8	An analytical model to evaluate reliability of cloud computing systems in the presence of QoS requirements	(Faragardi et al., 2013)	x					

Clúster	Clúster nivel 2 - "High (8)"
Consulta	Security Performance Financial metrics availability scalability agility experience latency privacy transparency Service
Subnodos	<ul style="list-style-type: none"> 📁 High (8) <ul style="list-style-type: none"> 📁 High (5) 📁 Ensure (2) 📁 Low (2) 📁 Mechanism (2) 📁 Model (2) 📁 Systems (2) 📁 Other Topics (1)

Artículos

Nro.	Nombre de artículo	Referencia
1.	Q-aware Quality of service based cloud resource provisioning	(S. Singh & Chana, 2015)
2.	A Runtime Quality Measurement Framework for Cloud Database Service Systems	(Klems et al., 2012)
3.	Quality Assessment of Software as a Service on Cloud Using Fuzzy Logic	(Baliyan & Kumar, 2013)
4.	Experience Availability Tail-Latency Oriented Availability in Software-Defined Cloud Computing	(Cai et al., 2017)v
5.	A novel trust model for selection of Cloud Service Provider	(M. K. Naseer et al., 2014)
6.	A Model-Based Deployment Framework of Integrated Public Cloud Service	(Z. Y. Wang et al., 2012)
7.	The interplay between timeliness and scalability in cloud monitoring systems	(Da Cunha Rodrigues et al., 2016)
8.	End-to-End QoS on Shared Clouds for Highly Dynamic, Large-Scale Sensing Data Streams	(Tolosana-Calasanz et al., 2012)

Relación entre subnodos

Nombre de artículo	Referencia							
		High (5)	Ensure (2)	Low (2)	Mechanism (2)	Model (2)	System (2)	Other Topics (1)
Q-aware Quality of service based cloud resource provisioning	(S. Singh & Chana, 2015)	x		x				
A Runtime Quality Measurement Framework for Cloud Database Service Systems	(Klems et al., 2012)	x					x	
Quality Assessment of Software as a Service on Cloud Using Fuzzy Logic	(Baliyan & Kumar, 2013)	x		x		x		
Experience Availability Tail-Latency Oriented Availability in Software-Defined Cloud Computing	(Cai et al., 2017)v		x			x		
A novel trust model for selection of Cloud Service Provider	(M. K. Naseer et al., 2014)							x
A Model-Based Deployment Framework of Integrated Public Cloud Service	(Z. Y. Wang et al., 2012)	x	x		x			
The interplay between timeliness and scalability in cloud monitoring systems	(Da Cunha Rodrigues et al., 2016)						x	
End-to-End QoS on Shared Clouds for Highly Dynamic, Large-Scale Sensing Data Streams	(Tolosana-Calasanz et al., 2012)	x			x			

Clúster	Clúster nivel 2 - “Security and Privacy Issues (8)”
Consulta	Access Control Privilege Management Data Geographic Political Data Integrity Data Privacy Data Loss Physical Environmental Security Proactive Threat Vulnerability Management Retention Disposition Security Management
Subnodos	<ul style="list-style-type: none"> 📁 Security and Privacy Issues (8) <ul style="list-style-type: none"> 📁 Cloud (5) 📁 Attributes (2) 📁 Computing (2) 📁 Confidentiality (2) 📁 Essential (2) 📁 Level (2) 📁 Service (2) 📁 Trust (2)

Artículos

Nro.	Nombre de artículo	Referencia
1.	Evaluating Security and Privacy in Cloud Services	(Abuhussein et al., 2016)
2.	An Approach to IdM with Privacy in the Cloud	(Werner et al., 2015)
3.	Cloud Computing The Issue of Service Quality An Overview of Cloud Service Level Management Architectures	(Motta et al., 2013)
4.	Security and Privacy in Cloud Computing	(Xiao & Xiao, 2013)
5.	Toward a framework for data quality in cloud-based health information system	(Almutiry et al., 1998)
6.	Trust management in cloud computing A survey	(Monir et al., 2015)
7.	A trust reputation architecture for virtual organization integration in cloud computing environment	(Bilecki et al., 2017)
8.	A trust model in cloud computing based on fuzzy logic	(Ritu & Jain, 2017)

Relación entre subnodos

Nro.	Nombre de artículo	Referencia								
			Cloud (5)	Attributes (2)	Computing (2)	Confidentiality (2)	Essential (2)	Level (2)	Service (2)	Trust (2)
1	Evaluating Security and Privacy in Cloud Services	(Abuhussein et al., 2016)	x						x	
2	An Approach to IdM with Privacy in the Cloud	(Werner et al., 2015)	x			x				
3	Cloud Computing The Issue of Service Quality An Overview of Cloud Service Level Management Architectures	(Motta et al., 2013)	x	x			x	x	x	
4	Security and Privacy in Cloud Computing	(Xiao & Xiao, 2013)	x	x		x				
5	Toward a framework for data quality in cloud-based health information system	(Almutiry et al., 1998)				x				
6	Trust management in cloud computing A survey	(Challagidad & Birje, 2018)		x			x			x
7	A trust reputation architecture for virtual organization integration in cloud computing environment	(Bilecki, Fiorese, & Matos, 2017)			x					
8	A trust model in cloud computing based on fuzzy logic	(Ritu & Jain, 2017)	x					x		

Clúster	Clúster nivel 2 - "Network (7)"
Consulta	Security network Wireless financial performance attributes cost usability reliability
Subnodos	<ul style="list-style-type: none"> 📁 Network (7) <ul style="list-style-type: none"> 📁 Response Time (3) 📁 Model (2) 📁 Multimedia (2) 📁 QoS Attributes in Total (2) 📁 Other Topics (2)

Artículos

Nro.	Nombre de artículo	Referencia
1.	Security issues and threats according to the attribute of cloud computing	(H. Lee et al., 2012)
2.	Fuzzy cloud service selection framework	(Tajvidi et al., 2014)
3.	A novel \TOPSIS\ evaluation scheme for cloud service trustworthiness combining objective and subjective aspects	(L. Lu & Yuan, 2018)
4.	Security & scalability architecture for next generation internet services	(Talukder & Prahalad, 2009)v
5.	A Novel QoS Prediction Approach for Cloud Service Based on Bayesian Networks Model	(Zhang et al., 2016)v
6.	Assessing Measurements of QoS for Global Cloud Computing Services	(Pedersen et al., 2011)
7.	A novel TOPSIS evaluation scheme for cloud service trustworthiness combining objective and subjective aspects	(L. Lu & Yuan, 2018)

Relación entre subnodos

Nro.	Nombre de artículo	Referencia					
			Response Time (3)	Model (2)	Multimedia (2)	QoS Attributes in Total (2)	Other Topics (2)
1	Security issues and threats according to the attribute of cloud computing	(H. Lee, Kim, Lee, & Won, 2012)			x		
2	Fuzzy cloud service selection framework	(Tajvidi et al., 2014)					x
3	A novel \TOPSIS\ evaluation scheme for cloud service trustworthiness combining objective and subjective aspects	(L. Lu & Yuan, 2018)	x			x	
4	Security & scalability architecture for next generation internet services	(Talukder & Prahalad, 2009)v					x
5	A novel TOPSIS evaluation scheme for cloud service trustworthiness combining objective and subjective aspects	(Zhang et al., 2016)v	x	x			
6	Assessing Measurements of QoS for Global Cloud Computing Services	(Pedersen et al., 2011)		x	x		
7	Novel topsis evaluation scheme for cloud service trustworthiness combining objective and subjective aspects	(L. Lu & Yuan, 2018)	x			x	

Clúster	Clúster nivel 2 - “Distributed Environments (5)”
Consulta	cost usability reliability security performance availability scalability
Subnodos	<ul style="list-style-type: none"> 📁 Distributed Environments (5) <ul style="list-style-type: none"> 📁 Trustworthy in Order to make a Decision (2) 📁 Workflow Scheduling Approaches in Cloud Computing Environments (2) 📁 Other Topics (1)

Artículos

Nro.	Nombre de artículo	Referencia
1.	Cost-aware challenges for workflow scheduling approaches in cloud computing environments: Taxonomy and opportunities .	(Alkhanak et al., 2015)
2.	A novel \TOPSIS\ evaluation scheme for cloud service trustworthiness combining objective and subjective aspects	(L. Lu & Yuan, 2018)
3.	Service replication taxonomy in distributed environments	(Mohamed, 2016)
4.	Cost-aware challenges for workflow scheduling approaches in cloud computing environments: Taxonomy and opportunities	(Alkhanak et al., 2015)
5.	A novel TOPSIS evaluation scheme for cloud service trustworthiness combining objective and subjective aspects	(L. Lu & Yuan, 2018)

Nota: El proceso de preselección y selección de artículos fue realizado por 3 investigadores en un lapso de 3 meses. Cada uno recibió los parámetros de revisión relacionados en el apartado Procesamiento manual. El proceso primario para eliminar estudios repetidos se realizó a través de la comparación de títulos en los archivos de Excel, generados por los repositorios a partir de sus metadatos. El procesamiento automático realizado con la herramienta Carrot2 permitió evidenciar la existencia de 2 estudios repetidos que contenían caracteres especiales en el título: “/” y “ “, como se puede evidenciar en la tabla anterior, para los artículos de las filas 1 y 2.

Después de dialogar sobre el hallazgo, con los investigadores involucrados en esta fase del proyecto, se llegó a la conclusión que no representa una amenaza a la validez de la investigación, toda vez que los resultados obtenidos de la revisión y procesamiento de los estudios, no afectan la cantidad y definición de los atributos de calidad identificados para la taxonomía. Por lo anterior, se decidió dar continuidad a la investigación sin afectar las fases anteriores de la presente investigación.

Relación entre subnodos

Nro.	Nombre de artículo	Referencia				
			Trustworthy in Order to make a decision (2)	Workflow Scheduling Approaches in Cloud Computing Environments (2)	Other Topics (1)	
1	Cost-aware challenges for workflow scheduling approaches in cloud computing environments: Taxonomy and opportunities .	(Alkhanak et al., 2015)		x		
2	A novel \TOPSIS\ evaluation scheme for cloud service trustworthiness combining objective and subjective aspects	(L. Lu & Yuan, 2018)	x			
3	Service replication taxonomy in distributed environments	(Mohamed, 2016)				x
4	Cost-aware challenges for workflow scheduling approaches in cloud computing environments: Taxonomy and opportunities	(Alkhanak et al., 2015)		x		
5	A novel TOPSIS evaluation scheme for cloud service trustworthiness combining objective and subjective aspects	(L. Lu & Yuan, 2018)	x			

Clúster	Clúster nivel 2 - "Survey (5)"
Consulta	security robustness QoS usability Service quality performance strongly interoperability
Subnodos	<ul style="list-style-type: none"> 📁 Survey (5) 📁 Systems (3) 📁 Applications (2) 📁 Cloud (2) 📁 Identifying (2)

Artículos

Nro.	Nombre de artículo	Referencia
1.	A Survey on Quality Assurance Techniques for Big Data Applications	(Zhang et al., 2017)
2.	A survey on quality of service in cloud computing	(Jelassi et al., 2017)
3.	A survey on reliability in distributed systems	(W. Ahmed & Wu, 2013)
4.	A survey on elasticity management in PaaS systems	(Muñoz-Escóí & Bernabéu-Aubán, 2017)
5.	A survey on cloud federation architectures Identifying functional and non-functional properties	(Assis & Bittencourt, 2016)

Relación entre subnodos

Nro.	Nombre de artículo	Referencia				
			Systems (3)	Applications (2)	Cloud (2)	Identifying (2)
1	A Survey on Quality Assurance Techniques for Big Data Applications	(Zhang, Zhou, Li, & Gao, 2017)	x	x		
2	A survey on quality of service in cloud computing	(Jelassi, Ghazel, & Saidane, 2017)			x	
3	A survey on reliability in distributed systems	(W. Ahmed & Wu, 2013)	x	x		
4	A survey on elasticity management in PaaS systems	(Muñoz-Escóí & Bernabéu-Aubán, 2017)	x			x
5	A survey on cloud federation architectures Identifying functional and non-functional properties	(Assis & Bittencourt, 2016)			x	x

Clúster	Clúster nivel 2 - "Distributed Systems (4)"
Consulta	security privacy confidentiality Cost distribution
Subnodos	<ul style="list-style-type: none"> 📁 Distributed Systems (4) <ul style="list-style-type: none"> 📁 Cost-aware Workflow Scheduling (2) 📁 Survey (2)

Artículos

Nro.	Nombre de artículo	Referencia
1.	A survey on data confidentiality and privacy in cloud computing	(Aloraini & Hammoudeh, 2017)
2.	Cost-aware challenges for workflow scheduling approaches in cloud computing environments: Taxonomy and opportunities .	(Alkhanak et al., 2015)
3.	A survey on reliability in distributed systems	(W. Ahmed & Wu, 2013)
4.	Cost-aware challenges for workflow scheduling approaches in cloud computing environments: Taxonomy and opportunities	(Alkhanak et al., 2015)

Relación entre subnodos

Nro.	Nombre de artículo	Referencia		
			Cost-aware Workflow Scheduling (2)	Survey (2)
1	A survey on data confidentiality and privacy in cloud computing	(Aloraini & Hammoudeh, 2017)		x
2	Cost-aware challenges for workflow scheduling approaches in cloud computing environments Taxonomy and opportunities	(Alkhanak et al., 2015)	x	
3	A survey on reliability in distributed systems	(W. Ahmed & Wu, 2013)		x
4	Costaware challenge for workflow scheduling approaches in cloud computing environments Taxonomy and opportunities	(Alkhanak et al., 2015)	x	

Clúster	Clúster nivel 2 - “Non-functional Requirements (3)”
Consulta	Security performance quality performance privacy trust
Subnodos	<ul style="list-style-type: none"> 📁 Non-functional Requirements (3) <ul style="list-style-type: none"> 📁 Approach (2) 📁 Violation (2) 📁 Web Service (2)

Artículos

Nro.	Nombre de artículo	Referencia
1.	Towards an Approach for Trustworthiness Assessment of Software as a Service	(Medeiros et al., 2017)
2.	A framework for IaaS-to-SaaS monitoring of BPEL processes in the Cloud Design and evaluation	(Grati et al., 2014)
3.	A trustworthy QoS-based collaborative filtering approach for web service discovery	(Lin et al., 2014)

Relación entre subnodos

Nro.	Nombre de artículo	Referencia			
			Approach (2)	Violation (2)	Web Service (2)
1	Towards an Approach for Trustworthiness Assessment of Software as a Service	(Medeiros et al., 2017)	x	x	
2	A framework for IaaS-to-SaaS monitoring of BPEL processes in the Cloud Design and evaluation	(Grati et al., 2014)		x	x
3	A trustworthy QoS-based collaborative filtering approach for web service discovery	(Lin et al., 2014)	x		x

Los 10 nodos siguientes del clúster de nivel 1 generado por Carrot2, agruparon solamente dos trabajos cada uno, como se aprecia en la Figura 42. Para efectos de organización de los trabajos relacionados en cada nodo, la Tabla 49 presenta el nombre del nodo, la consulta aplicada, el nombre de los trabajos y su referencia.

📁 Evaluation Model (20)	📁 Distributed Environments (5)
📁 Reliability (19)	📁 Survey (5)
📁 Availability (17)	📁 Distributed Systems (4)
📁 Service Provider (17)	📁 Non-functional Requirements (3)
📁 Service Selection (16)	📁 Adaptive Monitoring (2)
📁 Quality Attributes (15)	📁 Dependability (2)
📁 Measurement (12)	📁 Hybrid (2)
📁 Metric Model (12)	📁 Maintaining Data (2)
📁 Data Privacy (11)	📁 Maintenance (2)
📁 Software Quality (11)	📁 Nonfunctional (2)
📁 QoS Requirements (9)	📁 Software Quality Assurance (2)
📁 Aspects (8)	📁 Systematic Literature Review (2)
📁 Cloud Computing Systems (8)	📁 Trust is the most Prominent Problem (2)
📁 High (8)	📁 Workflow Scheduling (2)
📁 Security and Privacy Issues (8)	📁 Other Topics (15)
📁 Network (7)	

Figura 42. Clúster nivel 1 generado por Carrot2

Tabla 49. Nodos de clúster nivel 1 que relacionaron solamente 2 trabajos

Nodo	Consulta	Artículos	Referencias
Adaptative Monitoring (2)	QoS accountability Quality service performance monitoring	CloudRec A Mobile Cloud Service Recommender System Based on Adaptive QoS Management	(Tang & Yan, 2015)
		An adaptive monitoring framework for ensuring accountability and quality of services in cloud computing	(H. M. Khan et al., 2016)
Dependability (2)	dependability performance security QoS metrics	Differentiated Availability in Cloud Computing SLAs	(Undheim et al., 2011)
		Brokerage-based dependability integration in cloud computing services	(Abderrahim & Choukair, 2018)
Hybrid (2)	Attributes security control privacy performance QoS quality	A Hybrid Model for Ranking Cloud Services	(De Benedetti et al., 2015)
		Evolution of service composition based on QoS under the cloud computing environment	(Hongzhen et al., 2016)
Maintaining Data (2)	security privacy confidentiality performance Integrity maintaining	A survey on data confidentiality and privacy in cloud computing	(Aloraini & Hammoudeh, 2017)
		Trustworthy mechanisms for selecting cloud service providers	(Elizabeth et al., 2014)
Maintenance (2)	Reliability Security Responsiveness Tangibility Empathy Assurance Functional maintenance interoperability	Quality Model of Maintenance Service for Cloud Computing	(Z. Wang et al., 2015)
		A trust reputation architecture for virtual organization integration in cloud computing environment	(Bassil, 2012)
Nonfunctional (2)	Security performance quality performance privacy trust	Trustworthy mechanisms for selecting cloud service providers	(Elizabeth et al., 2014)
		Functional and Nonfunctional Quality in Cloud-Based Collaborative Writing An Empirical Investigation	(Kim et al., 2014)
Software Quality Assurance (2)	Usability Performance scalability elasticity efficiency metrics	Beyond simulation Composing scalability, elasticity, and efficiency analyses from preexisting analysis results	(S. Lehrig & Becker, 2015)
		Quality evaluation and best service choice for cloud computing based on user preference and weights of	(Choi & Jeong, 2014)

Nodo	Consulta	Artículos	Referencias
		attributes using the analytic network process	
Systematic Literature Review (2)	Transparency security performance QoS functionality	An Overview of Software Functionality Service A Systematic Literature Review	(Salleh et al., 2017)
		Cloud service evaluation method-based Multi-Criteria Decision-Making A systematic literature review	(Alabool et al., 2018)
Trust is the most Prominent Problem (2)	Performance quality service	A novel trust model based on SLA and behavior evaluation for clouds	(Z. Tan et al., 2016)
		MCTModel A multi-clouds trust model based on SLA in cloud computing	(Z.-H. Tan et al., 2017)
Workflow Sheduling (2)	Cost performance distribution	Cost-aware challenges for workflow scheduling approaches in cloud computing environments: Taxonomy and opportunities .	(Alkhanak et al., 2015)
		Cost-aware challenges for workflow scheduling approaches in cloud computing environments: Taxonomy and opportunities	(Alkhanak et al., 2015)

Clúster	Clúster nivel 2 - “Other Topics (15)”
Consulta	Performance Security Accountability Privacy agility assurance financial economics usability
Subnodos	<ul style="list-style-type: none"> 📁 Other Topics (15) <ul style="list-style-type: none"> 📁 Service (5) 📁 Security Issues (4) 📁 SLAs (3) 📁 Attributes (2) 📁 Elasticity (2) 📁 Framework (2) 📁 Instance (2) 📁 Mechanisms (2) 📁 Security and Privacy in Public Cloud Computing (2) 📁 Other Topics (1)

Artículos

Nro.	Nombre de artículo	Referencia
1.	Security and service assurance issues in Cloud environment	(Deshpande et al., 2018)
2.	Eliciting metrics for accountability of cloud systems	(Nuñez et al., 2016)
3.	Two-way Ranking Based Service Mapping in Cloud Environment	(Yadav & Goraya, 2018)
4.	Performance evaluation of cloud computing providers using fuzzy multiattribute group decision making model	(Grandhi & Wibowo, 2016)
5.	TcpRT Instrument and Diagnostic Analysis System for Service Quality of Cloud Databases at Massive Scale in Real-time	(Cao et al., 2018)
6.	A resiliency framework for an enterprise cloud	(Chang et al., 2016)
7.	EMSCLOUD - An evaluative model of cloud services cloud service management	(Techio & Misaghi, 2015)
8.	Survey on trust establishment in cloud computing	(G. Raj et al., 2014)
9.	An Integrated Approach for Specifying and Enforcing SLAs for Cloud Services	(Freitas et al., 2012)
10.	Scalability, elasticity, and efficiency in cloud computing A systematic literature review of definitions and metrics	(Sebastian Lehrig et al., 2015)
11.	A new trust evaluation approach for cloud computing environments	(Mrabet et al., 2017a)
12.	Principles of Elastic Processes	(Dustdar et al., 2011)
13.	Emscloud evaluative model of cloud services cloud service management	(Techio & Misaghi, 2015)
14.	A dynamic classification index to enhance data protection procedures in cloud-based environments	(Moghaddam et al., 2016)
15.	Dynamic IaaS Computing Resource Provisioning Strategy with QoS Constraint	(Ran et al., 2017)

Relación entre subnodos

Nro.	Nombre de artículo	Referencia														
			Service (5)	Security Issues (4)	SLAs (3)	Attributes (2)	Elasticity (2)	Framework (2)	Instance (2)	Mechanisms (2)	Security and Privacy in Public cloud Computing (2)	Other Topics (1)				
1	Security and service assurance issues in Cloud environment	(Deshpande, Sharma, Peddoju, & Abraham, 2018)	x	x												
2	Eliciting metrics for accountability of cloud systems	(Nuñez, Fernández-Gago, & Luna, 2016)						x		x						
3	Two-way Ranking Based Service Mapping in Cloud Environment	(Yadav & Goraya, 2018)				x										x
4	Performance evaluation of cloud computing providers using fuzzy multiattribute group decision making model	(Grandhi & Wibowo, 2016)														
5	TcpRT Instrument and Diagnostic Analysis System for Service Quality of Cloud Databases at Massive Scale in Real-time	(Cao et al., 2018)	x						x							
6	A resiliency framework for an enterprise cloud	(Chang, Ramachandran, Yao, Kuo, & Li, 2016)								x						
7	EMSCLOUD - An evaluative model of cloud services cloud service management	(Techio & Misaghi, 2015)		x												x
8	Survey on trust establishment in cloud computing	(G. Raj, Sarfaraz, & Singh, 2014)	x		x											
9	An Integrated Approach for Specifying and Enforcing SLAs for Cloud Services	(Freitas, Parlavantzas, & Pazat, 2012)	x		x						x					
10	Scalability, elasticity, and efficiency in cloud computing A systematic literature review of definitions and metrics	(Sebastian Lehrig, Eikerling, & Becker, 2015)				x										
11	A new trust evaluation approach for cloud computing environments	(Mrabet, Saied, & Saidane, 2017a)					x	x								
12	Principles of Elastic Processes	(Dustdar, Guo, Satzger, & Truong, 2011)									x					
13	Emscloud evaluative model of cloud services cloud service management	(Techio & Misaghi, 2015)		x												x
14	A dynamic classification index to enhance data protection procedures in cloud-based environments	(Moghaddam, Majd, Ahmadi, Khodadadi, & Madadipouya, 2016)														x
15	Dynamic IaaS Computing Resource Provisioning Strategy with QoS Constraint	(Ran, Yang, Zhang, & Xi, 2017)	x							x						

Apéndice B

Taxonomy details

TQACA - Accountability

Measurement	Accountability
Level	Root
Definition	This category contains attribute used to measure the properties related to the cloud service provider organization. These properties may be independent of the service being provided
Way to perform calculation	It depends on the children nodes
Additional information	Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non-functional

Measurement	Compliance
Level	Accountability
Definition	Standards, processes and policies committed to by the cloud service provider are followed
Way to perform calculation	Not defined by the authors
Additional information	Type: Quantitative Unit: Percentage Scope: Provider Category: Objective Requirement: Non-functional

Measurement	Contracting experience
Level	Accountability
Definition	Indicators of client efforts and satisfaction with the process of entering into the agreements required to use a service
Way to perform calculation	Not defined by the authors
Additional information	Type: Quantitative Unit: Ordered set Scope: User Category: Subjective Requirement: Non-functional

Measurement	Ease of doing business
Level	Accountability
Definition	Client satisfaction with the ability to do business with a cloud service provider
Way to perform calculation	Not defined by the authors
Additional information	Type: Qualitative Unit: Ordered set Scope: User Category: Subjective Requirement: Non-functional

Measurement	Governance
Level	Accountability
Definition	The processes used by the cloud service provider to manage client expectations, issues and service performance
Way to perform calculation	Not defined by the authors
Additional information	Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non-functional

Measurement	Provider certifications
Level	Accountability
Definition	The cloud service provider maintains current certifications for standards relevant to their clients' requirements. Identifies CSP compliance to standards and assesses CSP background, qualifications, and legitimacy in information security, privacy, healthcare, education, etc. Considerations include: (1) Is the CSP and Service certified?, (2) can consumers see standards, certifications, and reports from granting organizations including employee training?
Way to perform calculation	User can give weights to certifications according to its need eventually to rank providers.
Additional information	Type: Quantitative Unit: Numeric Scope: User Category: Objective Requirement: Non-functional

Measurement	Provider contract /SLA verification
Level	Accountability
Definition	<p>The cloud service provider makes a available to clients SLAs adequate to manage the service and mitigate risk of service failure.</p> <p>Identifies whether the running service conforms with the SLA. Consideration are: (1) Does the CSP consider complexity of the cloud architecture in SLA?, (2) Does SLA consider different impacts of service outage on different business natures?</p>
Way to perform calculation	Not defined by the authors
Additional information	Type: Qualitative Unit: Boolean Scope: Provider Category: Objective Requirement: Non-functional

Measurement	Provider personnel requirements
Level	Accountability
Definition	<p>The extent to which cloud service provider personnel have the skills, experience, education, and certifications required to effectively deliver a service.</p> <p>Assesses the CSP policies, strategies, and procedures to prevent, detect, and/or respond to malicious attack perpetrated on cloud service by a person with authorized access. Considerations include: (1) Does the CSP have a formalized insider threat program?, (2) Does the CSP perform background check on employees?</p>
Way to perform calculation	Not defined by the authors
Additional information	Type: Qualitative Unit: Ordered set Scope: Provider Category: Objective Requirement: Non-functional

Measurement	Provider supply chain
Level	Accountability
Definition	<p>The cloud service provider ensures that any SLAs that they must be supported by its suppliers are supported.</p> <p>Assesses the CSP policies, strategies, and procedures used to validate S&P legitimacy and compatibility of the purchased hardware and software that are used in clouds. Considerations include: (1) Does the</p>

	CSP purchase/lease hardware and software from a legitimate source? (2) Does the CSP purchase/lease hardware and software that are all tested, certified, and conform to standards?
Way to perform calculation	Not defined by the authors
Additional information	Type: Qualitative Unit: Boolean Scope: Provider Category: Objective Requirement: Non-functional
Measurement	Provider support
Level	Accountability
Definition	The extent to which the cloud service provider includes or makes available assistance to the client in their efforts to use the service, including answering questions about the service and working around or correcting any problems that may arise. Refers to the CSP's willingness to compensate cloud consumers for specific potential service failures. Considerations include: (1) Does the CSP provide insurance for the services?, (2) Does the CSP have more than one insurance plan?
Way to perform calculation	Not defined by the authors
Additional information	Type: Qualitative Unit: Ordered set Scope: User Category: Objective Requirement: Non-functional
Measurement	Customer support facility
Level	Accountability » Provider support
Definition	Type of support, response time for support and the charge for customer support are important factors which define customer support facility. Generally new users prefer a provider with a good support system. Some providers offer free customer support service, but mostly the providers charge accordingly. GoGrid provides free 24/7 phone support and free 24/7 premium support. Amazon AWS provides premium Support (Urgent - 1 hour, High - 4 business hours, Normal - 1 business day, Low - 2 business days). So to measure customer support facility metric, the metric is assumed as unordered set which may contain elements like Free 24/7 phone support, Urgent support, Basic support, Low support, Diagnostic tools etc.

Way to perform calculation	<p>We get the feedback for last one year which is done after each quarter of the year.</p> $CSE = \frac{TM}{50}$ <p>Where TM = Total Marks and</p> $TM = 1 * MQ1 + 2 * MQ2 + 3 * MQ3 + 4 * MQ4$ <p>'1' is the weight (multiplier) for 1 st Quarter '2' is the weight (multiplier) for 2nd Quarter '3' is the weight (multiplier) for 3rd Quarter '4' is the weight (multiplier) for 4th Quarter</p>
Additional information	<p>Type: Quantitative Unit: Numeric Scope: Application Category: Subjective Requirement: Non-functional</p>
Measurement Level	<p>Operating Systems Support Accountability » Provider support</p>
Definition	<p>Providers support different OS like Mac OS X, Windows, and Open SUSE Linux etc. It is possible that one provider support some OS and other provider supports some other OS like Windows Azure supports only Windows operating system, while GoGrid supports Windows server 2003/2008, Red hat Linux 5.1/5.4 etc. Different applications require different OS support.</p>
Way to perform calculation	<p>Not defined by the authors</p>
Additional information	<p>Type: Quantitative Unit: Ordered set Scope: Developer Category: Objective Requirement: Non-functional</p>
Measurement Level	<p>Platforms Supported Accountability » Provider support</p>
Definition	<p>Same as operating systems, different providers support different type of platforms. For example, CloudSigma supports Java, PHP, WinDev, Dot Net. While Firehost supports Ruby, Java, PHP, and Dot Net. Different application requires different platform support.</p>
Way to perform calculation	<p>Not defined by the authors</p>
Additional information	<p>Type: Quantitative Unit: Ordered set Scope: Developer</p>

	Category: Objective Requirement: Non-functional
Measurement	Sustainability
Level	Accountability
Definition	<p>Sustainability has three dimensions; economic, social and environmental impact of Cloud service provider [9]. Environmental impact can be measured using metrics like carbon emission, water use and resource consumption.</p> <p>Economic impact can be measured using metrics such as power cost of servers, storage, and networking facility cost and support cost. Social impact can be measured using metrics like economic development and socio-political stability.</p>
Way to perform calculation	<p>Measurement of all three impacts is complex. Some common metrics are PUE (power usage effectiveness), CUE (carbon usage effectiveness), CUPS (compute units per second), CADE (corporate average data center efficiency), DPPE (data center performance per energy), and WUE (water use efficiency).</p> <p>Some providers use their own measurement and auditing tools.</p>
Additional information	<p>Type: Quantitative Unit: Descriptive Scope: Provider Category: Objective Requirement: Non-functional</p>
Measurement	Technology Change
Level	Accountability
Definition	<p>Measures the CSP readiness to cope with the S&P issues that may occur as a result of technology evolution or vanishing. Considerations include: (1) Can the CSP change security features in case of technology emergence/ obsolescence without affecting the service?, (2) Does the CSP guide consumers to make educated choices if technology or S&P attributes or risks change?</p>
Way to perform calculation	Not defined by the authors
Additional information	<p>Type: Qualitative Unit: Boolean Scope: Provider Category: Objective Requirement: Non-functional</p>

TQACA - Agility

Measurement	Agility
Level	Root
Definition	Indicates the impact of a service upon a client's ability to change direction, strategy, or tactics quickly and with minimal disruption.
Way to perform calculation	It depends on the children nodes
Additional information	Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non-functional

Measurement	Adaptability
Level	Agility
Definition	Reflects on the ability of the cloud computing provider to cope with unexpected organizations' requests by adjusting or modifying changes in its services. Adaptability is seen increasingly as an important factor for improving an organization's efficiency and achieving economic success.
Way to perform calculation	It is measured with two metrics; Coverage of Variability (CoV) and Completeness of Variant Set (CoM). Both metrics are explained in Availability and Reusability. The Adaptability is derived from the other two metrics: Adaptability = $W_{CoV} \cdot CoV + W_{CoM} \cdot CoM$ where W_{CoV} , W_{CoM} are the weights for each metric which the sum is 1. The weight for each metric is assigned based on the importance of each metric. The range of Adaptability is 0 ... 1 and higher value of Adaptability shows that the web services by incorporating other services in with is more easily and efficaciously customized to service users- specific need.
Additional information	Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non-functional

Measurement	Flexibility
Level	Agility
Definition	The ability to add or remove predefined features from a service
Way to perform calculation	Related technical issues can be given emphasis on flexibility, capacity adjustments without loss of performance.

	$FLEX = \frac{\text{current resources} + \text{changes}}{\text{performance}}$
Additional information	Type: Quantitative Unit: Numeric Scope: Developer Category: Objective Requirement: Non-functional
Measurement Level	Portability Agility
Definition	The ability of a client to easily move a service from one cloud service provider to another with minimal disruption. Refers to the ability of a service provider to move the data from one cloud system to another without degrading on performance. So, a service provider is said portable over a period T if it remains reliable and time effective under the circumstance of jumping from one cloud system to another.
Way to perform calculation	Portability: It can be formulated as follows: Portability of service = Degree to which the service or CS is portable to other platforms $\text{Portability} = \frac{\text{No. of compatible platforms}}{\text{total no. of platforms}}$
Additional information	Type: Quantitative Unit: Numeric Scope: Developer Category: Objective Requirement: Non-functional
Measurement Level	Responsiveness Agility
Definition	Customers demand the response from service timely. That's why responsiveness of cloud service is pointed out as important attribute to ensure quality. Can be defined as the degree to which the cloud service provider is ready and able to help customers by providing prompt timely services. Responsiveness (RESP) is the promptness with which cloud services perform a request during a time Interval. Instead of a maximum possible time from a system perspective, we use the maximum acceptable time from an end-user's viewpoint to formulate

	<p>responsiveness. For cloud services, a measurement from an end-user's viewpoint is better than the one from a system perspective.</p> <p>Few cloud services currently mention responsiveness in their SLAs. As cloud services advocate computation, in a general meaning, as a service, where a quick response indicates high performance, consumers value an accurate and prompt service.</p>
Way to perform calculation	$\tau = 1 - \frac{f_{i=1}^n(t_i)}{t_{\max}}$ <p>where $0 \leq \tau \leq 1$ represents responsiveness, t denotes the time between the submission and the completion of the ith request, t_{\max} is a parameter that specifies the maximum acceptable time to complete a request ($t_{\max} \geq t_i$), n is the number of requests issued in an operational period, and f is a function that measures the central tendency of a set of data, such as the mean and the median. The closer the value of τ is to 1, the better the responsiveness.</p>
Additional information	<p>Type: Quantitative Unit: Numeric Scope: User Category: Objective Requirement: Non-functional</p>

Measurement	Elasticity
Level	Agility » Responsiveness
Definition	<p>The ability of a cloud service provider to adjust its resource consumption for a service at a rapid enough rate to meet client demand.</p> <p>Elasticity can be understood by its two fundamental elements: Time and Cost. Time means how much time a service provider takes to provision or de-provision resources and cost means whether service provider charges on per hour basis or per minute basis.</p>
Way to perform calculation	<p>Let w_1 and w_2 be weights of time and cost respectively i.e. importance of time and cost for a user with $w_1 + w_2 = 1$.</p> <p>elasticity = w_1*time + w_2*cost</p> <p>Another formula:</p> $\varepsilon = \frac{\sum_{i_1=1}^n r_{i_1}}{\sum_{i_2=1}^n r_{i_2}}$ <p>where $0 \leq \varepsilon \leq 1$ represents elasticity, r_{i_1} and r_{i_2} denote the amount of</p>

	resources allocated and requested in the i th request, respectively, and n is the number of requests issued in an operational period. The closer the value of ϵ is to 1, the higher the elasticity.
Additional information	Type: Quantitative Unit: Ordered set Scope: Developer Category: Objective Requirement: Non-functional

Measurement	Mean Time To Quality Repair
Level	Agility » Responsiveness » Elasticity
Definition	(MTTQR) We derive the mean time to quality repair metric from EMadapt, the measure how quickly a system can adapt to workload changes. It is a measure for elasticity and depends on a workload delta, i.e., the increase/decrease between two workloads. MTTQR defines the mean time a system needs to reestablish its SLOs when the workload increases/decreases for a defined workload delta specified as factor (real number). Hence, MTTQR is measured in time units. Since it defines a mean time, MTTQR is specific for a specified time frame in which the mean is calculated.
Way to perform calculation	For example, with the same workload increase factor of 1.2, a perfectly elastic system Will adapt itself to the increasing workload within zero time, i.e., $MTTQR_{SA;1d}(1.2 \text{ req/s}) = 0 \text{ min}$. A less elastic system, e.g., will need a mean time of 10 min (calculated over one day) to adapt itself to increasing workload after it detects the workload increase, i.e., $MTTQR_{SB;1d}(1.2 \text{ req/s}) = 10 \text{ min}$.
Additional information	Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non-functional

Measurement	Number of SLO violations
Level	Agility » Responsiveness » Elasticity
Definition	(NSLOV) The number of SLO violations in a defined time interval is derived from EMviol. This metric measures elasticity of a system. The workload delta is specified as a factor (real number) as well. NSLOV reflects how often a system violates its SLOs when workload changes at a given rate, measured as a real number.
Way to perform calculation	For example, with a workload increase factor of 1.2, a perfectly elastic system would have 0 SLO violations per request, i.e., $NSLOV_{SA}(1.2 \text{ req/s}) = 0$. In contrast, a non-elastic system will violate its SLO for each request, i.e., $NSLOV_{SA}(1.2 \text{ req/s}) = 1$.

Additional information	Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non-functional
------------------------	--

Measurement	VM Shutdown Time
Level	Agility » Responsiveness » Elasticity
Definition	Time consumed to shut down and deallocate VM (s)
Way to perform calculation	Not defined by the authors
Additional information	Type: Quantitative Unit: Numeric Scope: Developer Category: Objective Requirement: Non-functional

Measurement	VM Startup Time
Level	Agility » Responsiveness » Elasticity
Definition	Time consumed to allocate, boot up, and configure VM (s)
Way to perform calculation	Not defined by the authors
Additional information	Type: Quantitative Unit: Numeric Scope: Developer Category: Objective Requirement: Non-functional

Measurement	Interaction
Level	Agility » Responsiveness
Definition	Degree to which a cloud service respond to customers' request in appropriate way
Way to perform calculation	Not defined by the authors
Additional information	Type: Quantitative Unit: Numeric Scope: User Category: Objective Requirement: Non-functional

Measurement	Timeliness
-------------	------------

Level	Agility » Responsiveness
Definition	<p>Degree to which the speed of response of a cloud service meets customer needs.</p> <p>Is an ability with which a cloud monitoring system is able to supply information in time when users need to access it. It measures the total time taken to complete a task. This is measured as delay, or latency, or time to complete. This metric also measures the start time and deadline for a task.</p>
Way to perform calculation	$T_L(i) = T_{wait}(i) + T_{exec}(i) + T_{trans}(i)$ <p>where $T_L(i)$ represents the total completion time, $T_{wait}(i)$ is the waiting time, $T_{exec}(i)$ is the task execution time and $T_{trans}(i)$ indicates the transmission time of the i-th task during VM migration. The delay time T_{Dij} of user i to access resources on cloud j is determined by:</p> $T_{Dij} = T_{Rij} + T_{Drr} + T_{Nij}$ <p>where T_{Rij} is the response time from the resources, T_{Drr} is the delay time in resource redirection caused by the redirection overhead for the purpose of load balancing, and T_{Nij} is the network time the request takes to travel from client i to cloud j.</p>
Additional information	<p>Type: Quantitative Unit: Numeric Scope: Developer Category: Objective Requirement: Non-functional</p>
Measurement	Scalability
Level	Agility
Definition	<p>Is a measure of an application system's ability to provide increased throughput, reduced response time and/or support more users when hardware resources are added without modification in cost-effectively.</p> <p>Refers to the capability of the cloud computing supplied by the cloud computing provider to handle range of request made simultaneously by the applications. The ability to accommodate extra demand in resources is also an essential part of the flexibility required from the cloud computing provider.</p> <p>We distinguish two types of scalability: Vertical scalability and Horizontal scalability.</p> <p>Vertical scalability: Scaling vertically or scaling up means to add more resources to a given platform node, like CPU cores or memory in a way</p>

	<p>that the platform node can handle a larger workload.</p> <p>Horizontal scalability: Scaling horizontally or scaling out means adding new nodes to a cluster or distributed system in a way that the entire system can handle bigger workloads.</p> <p>Considerations include: (1) Does the CSP guarantee availability while scaling up/down?, (2) Does the CSP allow scaling to different geographical zones?</p>
Way to perform calculation	<p>If an application delivers a good performance with N resources for X users, and if its performance remains the same by multiplying resources by 2 when the number of users is multiplied by 2, then this application has a good scalability. If this ratio remains constant as the two factors increase, the scalability will be linear. This is optimal. But it is more likely that some components will not be up to the desired scalability, and therefore that performance stabilizes at a certain threshold, despite the addition of additional resources.</p> <p>Coverage of Scalability. This metric measures the average amount of allocated resources among the amount of requested resources. This can be computed as:</p> <p>(summation i=1 to K (Amount of requested resources of ist request/ Amount of allocated resources of ist request)) / K</p>
Additional information	<p>Type: Quantitative Unit: Numeric Scope: Developer Category: Objective Requirement: Non-functional</p>
Measurement Level	<p>Customizability Agility » Scalability</p>
Definition	<p>It can be formulated as follows: $\frac{N_d}{N_d + N_s}$</p> <p>Nd: Number of dynamic changes in a Cloud service with respect to a workload. Ns: Number of static changes in a Cloud service with respect to a workload.</p>
Way to perform calculation	<p>It can be formulated as follows: $\frac{N_d}{N_d + N_s}$</p> <p>Nd: Number of dynamic changes in a Cloud service with respect to a workload.</p>

	Ns: Number of static changes in a Cloud service with respect to a workload.
Additional information	Type: Quantitative Unit: Numeric Scope: Developer Category: Objective Requirement: Non-functional

Measurement	Scalability Range
Level	Agility » Scalability
Definition	(ScR) Based on EMscale, we define scaling range as a scalability metric that reflects a cloud computing system's ability to achieve its SLOs in a certain workload range, e.g., a range of request rates. For each single workload within this range, the system achieves its SLOs. The workload range is defined as a maximum workload.
Way to perform calculation	For example, a perfectly scalable system SA can scale up to a infinite request rate, i.e., ScRSA = req/min. A less scalable system, for example, scales up to a request rate of 112 requests per minute, i.e., ScRSB = 112 req/min.
Additional information	Type: Quantitative Unit: Numeric Scope: Developer Category: Objective Requirement: Non-functional

Measurement	Scalability Speed
Level	Agility » Scalability
Definition	(ScS) Based on EMrate, we define scalability speed ScS as a workload range with a maximal change rate in which a system can scale. This metric is a scalability metric which additionally considers the rate at which a system can scale. That is, the metric defines that a system is able to achieve its SLOs at each time when the workload changes at a maximal changing rate. The rate is defined by a maximum workload and an increase rate.
Way to perform calculation	For example, a scalable system can scale up to a request rate of 112 req/min with a linear increase rate of 1 additional requests per month, i.e., ScSSA = (112 req/min; 1 req/month).
Additional information	Type: Quantitative Unit: Numeric Scope: Developer Category: Objective Requirement: Non-functional

Measurement	Virtual infrastructure automation
Level	Agility » Scalability
Definition	Auto scaling, Deployment, configuration
Way to perform calculation	Not defined by the authors
Additional information	Type: Quantitative Unit: Numeric Scope: Developer Category: Objective Requirement: Non-functional

Measurement	Virtualization Technique
Level	Agility » Scalability
Definition	Unlike platforms, generally a provider supports only one virtualization platform. But each virtualization technique has its own advantages. If any one does not require advance features it may seek Hyper-V. Xen is Linux based and its management and administration is difficult but it has its advantage for experienced Linux user. VMware provides robust features. There are many vendors with application to fulfil the need in Vsphere. Hyper-V is freeware but one has to buy windows server. So choice of hypervisor very much depends on application's requirement
Way to perform calculation	All the VM groups required by a service are as follows: $vmg_{s,d} = (v_1, \dots, v_{\#vmg})$ The index d to represent a specific deployment configuration for service s. The resource space is defined using the cloud resources required by a service: $rc_s = \{vmg_{s,1}, \dots, vmg_{s,\#rc}\}$
Additional information	Type: Quantitative Unit: Descriptive Scope: Developer Category: Objective Requirement: Non-functional

TQACA - Assurance

Measurement	Assurance
Level	Root
Definition	This category includes key attributes that indicate how likely it is that the service will be available as specified

Way to perform calculation	It depends on the children nodes
Additional information	Type: Quantitative Unit: Numeric Scope: User Category: Objective Requirement: Non-functional
Measurement Level	Availability Assurance
Definition	<p>Availability is the degree to which a system or component is operational and accessible when required for use.</p> <p>Availability uses average response time (referred to RES) to represent.</p> <p>Defines the percentage of time when a resource is ready for immediate use and customers can access it.</p> <p>Considerations include: (1) Does the CSP ensure fair availability for all tenants on a physical machine?, (2) Does the CSP make best effort to protect isolated VMs that share the same partition from DoS attacks?</p>
Way to perform calculation	<p>% Time service is up: $up = t / ts$</p> <p>“up” represents metric of uptime percentage “t” and “ts” denote the uptime and the total time of an operational period, respectively The closer the value of “up” is to 1, the higher the uptime percentage.</p> <p>Another formula: RES_n expresses the average response time of the nth tasks that have been accepted and made successful responses in a single transaction, the definition of RES_i is followed as formula (2). RES_i expresses the response time of i-th task in the n-th tasks sent, and is equal to the difference of receiving time and the response time. N is the total number of tasks sent and responded successfully.</p> <p>$RES_n = \text{summation } i=1 \text{ to } N \text{ } RES_{in} / N$</p> <p>Another Formula: Availability = $T - TN / T$ where T is the concerned time period, and TN is the total time for which the resource is down thus not available.</p> <p>Availability can also be expressed in terms of average downtime per</p>

	week, month or year or as total downtime for a given week, month or year.
Additional information	Type: Quantitative Unit: Numeric Scope: Developer Category: Objective Requirement: Non-functional

Measurement	Down Time
Level	Assurance » Availability
Definition	The time for which services are unavailable to the service user. The downtime should be minimized and if there arises a need to the CSP to get the system down for an upgrade, repair or maintenance then that should preferably be pre-planned.
Way to perform calculation	Number of planned outages in peak hours is represented by "A". Mathematical modeling of the said description is given below: Equation given below is used to calculate the number of planned peak hours downtimes entered by the user. $A = \sum_{i=1}^n D_{pph(i)}$
Additional information	Type: Quantitative Unit: Numeric Scope: Developer Category: Objective Requirement: Non-functional

Measurement	Failure Rate
Level	Assurance » Availability
Definition	Average number of failures for one VM in one hour
Way to perform calculation	Not defined by the authors
Additional information	Type: Quantitative Unit: Numeric Scope: Developer Category: Objective Requirement: Non-functional

Measurement	Jitter
Level	Assurance » Availability
Definition	Is defined as a variation in the delay of received packets expressed in milliseconds.

Way to perform calculation	$J = J' + (D(i-1, i) - J') / 16$ <p>So the jitter J is calculated continuously every time a Ping packet is received, based on the previous jitter value, J', and the value of D(i,j) which is the difference in Ping times between the i'th and j'th packets.</p>
Additional information	Type: Quantitative Unit: Numeric Scope: Developer Category: Objective Requirement: Non-functional

Measurement	Latency
Level	Assurance » Availability
Definition	Is a difference between the moment when a first packet bit passed the input checkpoint and that when the last packet bit passed the output checkpoint at the other end of a channel. It depends on several parameters, e.g., packet length, network latency, and throughput of a provider's network.
Way to perform calculation	Latency: It can be formulated as follows: Latency = Time of input a Cloud workload - Time of output produced with respect to that Cloud workload
Additional information	Type: Quantitative Unit: Numeric Scope: Developer Category: Objective Requirement: Non-functional

Measurement	Response time
Level	Assurance » Availability
Definition	<p>An indicator of the time between when a service is requested and when the response is available. Response time depends on infrastructure as well as application for which request has been generated. But when new users are going to use Cloud services, they can measure response time of particular Cloud service providers by some benchmark.</p> <p>This is usually measured by amount of time that the cloud computing provider takes to respond to the user's requirement. The service response time also normally considers the average and the maximum time to respond as agreed with the service provider.</p> <p>Refers to the time interval in milliseconds between the service consumer sending a message and the response is received.</p>

Way to perform calculation	<p>$Q_i(s)$ is the average response time of service s</p> <p>For example, collected service response times are 0ms, 7ms, 15ms, 22ms, 35ms, and 50ms, where 0ms represents no response, the service access is not successful, and the average response time is 21.5ms. Consequently 7ms, 15ms, 22ms are normal response time, while 35ms and 50ms are longer than the average time, thus representing abnormal response times.</p> <p>Another formula:</p> $T_{avg} = \sum_{i=1}^{C_s} T_i / C_s$ <p>The successful invocation counts (C_s).</p> <p>The response time of unsuccessful invokes is not considered in the computation of average response time to avoid the influence of extreme large response time caused by unsuccessful invokes.</p>
Additional information	<p>Type: Quantitative Unit: Numeric Scope: Developer Category: Objective Requirement: Non-functional</p>

Measurement	Service Time
Level	Assurance » Availability
Definition	It is the expected time S needed to execute a VM and it can be obtained as a function of the time S_l needed to locally execute a VM and the time S_r needed to execute a VM into the federated clouds.
Way to perform calculation	<p>S_r can be computed as the ratio between the expected number of running VMs and the expected throughput of transition T_{serv}:</p> $S_l = \frac{E[r_X^u]}{Thr(T_{serv})}$ <p>When the VM multiplexing is not allowed, such a value is equal to $1/\mu$. S_r depends on the federated cloud quality level (q_f) and it is given by</p> $S_r = \frac{1}{\mu \cdot q_f}$ <p>To obtain the expression of S, the terms S_l and S_r have to be weighted summed with respect to the conditional probability of being locally or remotely accepted given that the request is not dropped. Defining the reward functions that allow us to calculate the expected probabilities to be locally or remotely accepted:</p>

	$r^l = (P_{run}^{\#} < M \mid P_{queue}^{\#} < Q) ? 1 : 0,$ $r^r = (P_{run}^{\#} = M \ \& \ P_{queue}^{\#} = Q \ \& \ P_{send}^{\#} < D) ? a_f : 0$ <p>we can derive the following expression:</p> $S = S_l \cdot \frac{E[r_X^l]}{E[r_X^l] + E[r_X^r]} + S_r \cdot \frac{E[r_X^r]}{E[r_X^l] + E[r_X^r]}$
--	--

Additional information	Type: Quantitative Unit: Numeric Scope: Developer Category: Objective Requirement: Non-functional
------------------------	---

Measurement	Service use rate
Level	Assurance » Availability
Definition	The ratio between the cloud service's time to the final time used
Way to perform calculation	Not defined by the authors

Additional information	Type: Quantitative Unit: Numeric Scope: Developer Category: Objective Requirement: Non-functional
------------------------	---

Measurement	Successability
Level	Assurance » Availability
Definition	Number of responses divided by number of requests
Way to perform calculation	Number of responses divided by number of requests

Additional information	Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non-functional
------------------------	--

Measurement	Time Efficiency
Level	Assurance » Availability
Definition	A service provider is said time effective in a given time if it delivers the completed service request within the promised response time in SLA.

	The probability of a service provider to be time effective is measured by the ratio of the number of times the promised response time is respected to the number of service requests successfully completed.
Way to perform calculation	Not defined by the authors
Additional information	Type: Quantitative Unit: Numeric Scope: Developer Category: Objective Requirement: Non-functional

Measurement	Up Time
Level	Assurance » Availability
Definition	The time of the year when the services of the CSP is available to the users. The trustor prefers the cloud service provider that offers the best up time in the year with respect to its efficient availability to the users. Uptime clearly defines maximum time when services are available. Efficiency of any CSP can be determined from its up time.
Way to perform calculation	We are picking up only 50 random days because it is very difficult to enter all values for 365 days of a year. $UT = \frac{TA}{24}$ Equation given above clearly shows the uptime points. We have selected 50 random days in a year and checked that how many maximum hours' services were available.
Additional information	Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non-functional

Measurement	Waiting Time
Level	Assurance » Availability
Definition	It is the expected time W spent by users in the queue (either the system or the upload queue).
Way to perform calculation	Using Little's law and computing the expected number of users in the queues through the following reward functions: $r^{q1} = P_{queue}^{\#}$ $r^{q2} = P_{send}^{\#}$

	<p>we can derive the waiting time expression as:</p> $W = \frac{E[r_X^{q1}] + E[r_X^{q2}]}{Thr(T_{arr}) \cdot A}$ <p>where Thr(T_{arr}) is the expected throughput of transition T_{arr}.</p>
Additional information	<p>Type: Quantitative Unit: Numeric Scope: User Category: Objective Requirement: Non-functional</p>

Measurement	Dependability
Level	Assurance
Definition	To what extent the cloud service could be dependable and under which conditions.
Way to perform calculation	Measurable QoS Parameters: availability time, reliability level (high/medium/low).
Additional information	<p>Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non-functional</p>

Measurement	Maintainability
Level	Assurance
Definition	Maintainability refers to the ability for the cloud service provider to make modifications to the service to keep the service in working conditions.
Way to perform calculation	Maintainability is evaluated through Mean Time To Recover (MTTR).
Additional information	<p>Type: Quantitative Unit: Numeric Scope: Developer Category: Objective Requirement: Non-functional</p>

Measurement	Recoverability
Level	Assurance
Definition	<p>Recoverability is the degree to which a service is able to quickly resume a normal state of operation after an unplanned disruption.</p> <p>Measures service readiness to recover and protect the cloud service in</p>

	the event of a disaster. Considerations include: (1) Does the CSP offer a disaster recovery plan? (2) Is the recovery process automated?
Way to perform calculation	Not defined by the authors
Additional information	Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non-functional
Measurement Level	Reliability Assurance
Definition	Can be defined as the degree to which a cloud service is available, complete, correct, continuous, stable and consistent. Reliability reflects measures of how a service operates without failure under given conditions during a given time period. Reliability uses the task success rate (SUn) to represent. Considerations include: (2) Does the CSP regularly check S&P of the composite services?, (2) Does the CSP have a clear outsourcing policy?
Way to perform calculation	MTBF (mean time between failures). MTBF = Total time / Number of failures Another formula: N_{suc}^n represents the number of the n-th tasks requested and completed successfully in a single transaction. $SU_n = N_{suc}^n / N_{tot}^n$ Another formula: $\text{Reliability} = P_r[V] \times P[\overline{T}_F]$ $= \left(1 - \frac{nf}{n}\right) P[\overline{T}_F]$ where Pr[V] is the probability of violation, nf is the number of users who have experienced a failure, n is the number of users in a concerned time period, and P[TF] is the promised mean time to failure. Another formula:

	$Re = T_{reliable} + T_{total}$ <p>Where T_{reliable} is total work time of service instance, and T_{total} is total work time of all service</p>
Additional information	Type: Quantitative Unit: Scale Scope: Provider Category: Objective Requirement: Non-functional
Measurement	Completeness
Level	Assurance » Reliability
Definition	Degree to which a cloud service supports all the tasks and objectives specified by the customer
Way to perform calculation	<p>CoM = Summation i = 1 to n (Number of variants supported in variant point Total number of potential variants in variant point)/ n</p> <p>where n is the number of variation points supported in the service, the denominator is the total number of potential variants in each variation point, and the numerator is the sum of the number of variants in each variation point</p>
Additional information	Type: Quantitative Unit: Numeric Scope: User Category: Objective Requirement: Non-functional
Measurement	Continuity
Level	Assurance » Reliability
Definition	Degree to which a cloud service is provided under all circumstances, including mitigate risks to a reasonable level and recover business after suspension
Way to perform calculation	<p>Continuity can be divided into 3 metrics, which are shown next.</p> <p>1. Incidence of accidents (IoA). Objective</p> <p>$x = IoA, 0 \leq x \leq 1$ a = the number of accidents b = the number of allowed accidents in the service agreement</p> <p>when $a < b$, $x = 1 - a / b$ when $a \geq b$, $x = 0$</p> <p>The closer the value of “x” is to 1, the better the IoA.</p>

	<p>2. Repair rate of accidents (RRoA). Objective</p> <p>a = the number of accidents repaired within required time in service agreement b = the total number of accidents within a service period</p> <p>r = a / b r = RRoA, 0 <= r <= 1</p> <p>The closer the value of “r” is to 1, the better the RRoA.</p> <p>3. Readiness of emergency preparedness for key businesses (RoEPKB). Subjective</p> <p>k = v / 2 k = RoEPKB, 0 <= k <= 1</p> <p>v = 2 when service provider has established comprehensive mechanism on emergency preparedness for key businesses and can fulfill the specification well v = 1 when service provider has established mechanism on emergency preparedness for key businesses, but cannot fulfill the specification very well v = 0 when there is no mechanism on emergency preparedness for key businesses</p> <p>The closer the value of “k” is to 1, the better the RoEPKB.</p>
Additional information	Type: Quantitative Unit: Scale Scope: User Category: Objective Requirement: Non-functional
Measurement	Correctness
Level	Assurance » Reliability
Definition	Degree to which a cloud service uses the correct process and produces the correct results. Measures service readiness with pre-configured responses and actions to face S&P failures. This includes automatic detection, prevention and response to failures. Considerations include: (1) Does the CSP make the best effort to identify triggers that lead to abnormal behavior in the cloud?, (2) Does the CSP have tools to automatically prevent S&P risks?
Way to perform	It can be formulated as follows:

calculation	<p>Accuracy : $(\text{expected CS} - \text{expected CS} - \text{observed CS}) / \text{expected CS}$</p> <p>Completeness : total existing CSs/total requested CSs</p> <p>Defects/CS : Number of Defects reported/CS</p>
Additional information	<p>Type: Quantitative</p> <p>Unit: Numeric</p> <p>Scope: Provider</p> <p>Category: Objective</p> <p>Requirement: Non-functional</p>

Measurement	Coverage of Failure Recovery
Level	Assurance » Reliability
Definition	(CFR) metric measures the ratio of rectifying failures at a specific period of time
Way to perform calculation	<p>This can be computed as:</p> <p>CFR = $1 - \frac{\text{number of failure not rectified}}{\text{total number of failure}}$</p> <p>where the denominator is the total number of failures induced by faults. The range is 0 ... 1 and higher value indicates that the Web service has higher recoverability</p>
Additional information	<p>Type: Quantitative</p> <p>Unit: Numeric</p> <p>Scope: Provider</p> <p>Category: Objective</p> <p>Requirement: Non-functional</p>

Measurement	Coverage of Fault Tolerance
Level	Assurance » Reliability
Definition	(CFT) metric measures the ratio of enduring the happening of a fault without a failure
Way to perform calculation	<p>This can be computed as:</p> <p>CFT = $1 - \frac{\text{number of faults becoming failure}}{\text{total number of faults occurred}}$</p> <p>where the denominator is the total number of faults identified and the numerator is the number of faults which cause failures. The range is 0 . . . 1 and higher value indicates that the services has higher durability about various faults</p>
Additional information	<p>Type: Quantitative</p> <p>Unit: Numeric</p>

	Scope: Provider Category: Objective Requirement: Non-functional
--	---

Measurement	Packet Loss
Level	Assurance » Reliability
Definition	Defined as the percentage ratio of packets sent or received which did not reach their destination to the total number of packets transmitted over a specific time interval
Way to perform calculation	Not defined by the authors
Additional information	Type: Quantitative Unit: Percentage Scope: Provider Category: Objective Requirement: Non-functional

Measurement	Punctuality
Level	Assurance » Reliability
Definition	A service provider is said punctual if it remains time effective over a period T.
Way to perform calculation	Not defined by the authors
Additional information	Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non-functional

Measurement	Reputation
Level	Assurance » Reliability
Definition	Measures trustworthiness of a Cloud provider. It is based on the users' experience for service providers. Different users may have different opinions for the same provider
Way to perform calculation	Reputation of a provider can be calculated taking average of rank assigned to a provider by different users. It is assumed users ranks the providers in a given range. Reputation = $(\text{summation } i=1 \text{ to } n \text{ of } r_i) / n$ where r_i is rank of reputation assigned by user i to provider and n is the number of times a provider has been ranked

Additional information	Type: Quantitative Unit: Numeric Scope: User Category: Objective Requirement: Non-functional
Measurement Level	Reusability Assurance » Reliability
Definition	<p>The principle for defining this attribute is explicated by both service provider's and consumer's view. Typically, web services providers in cloud induce the business model pay per use. If web service renders functionalities which meet the consumers requirements, then the number of service consumers and return on investment (ROI) is consequently increased. Th is attribute assess, if functionalities provided by services are common to the requirements defined by service consumers. Hence, the reusability is considered as an indicator which can disclose ROI of service providers.</p> <p>Service consumers evaluate the service based on the number of subscribers and the number of provided features needed by service consumers. These factors indirectly indicate that quantitative values, which reveal supporting commonality, are important to service consumers. Understandability is assessed with Readability of Service (RoS) which measures the quality of services description. Publicity is measured with Awarability of Service (AoS) which measures how efficiently newly deployed services can be aware of by service consumers.</p>
Way to perform calculation	<p>Reusability is measured using four metrics: Readability, Publicity, Coverage of variability and Commonality feature.</p> <p>Reusability = $W_{UoS} \cdot UoS + W_{AoS} \cdot AoS + W_{CoV} \cdot CoV + W_{CF} \cdot CF$</p> <p>where $W_{UoS}, W_{AoS}, W_{CoV}, W_{CF}$ are the weights for each metric which the sum is 1. The weight for each metric is assigned based on the importance of each metric. The range of Reusability is 0 ... 1 and higher value of Reusability indicates that the web services is more reusable.</p>
Additional information	Type: Quantitative Unit: Numeric Scope: Developer Category: Objective Requirement: Non-functional
Measurement Level	Stability Assurance » Reliability

Definition	The ability that a cloud service is continuously and steadily available. Stability is the variance of response time.
Way to perform calculation	$Q_s(s) = \sqrt{\frac{\sum_{i=1}^N (Q_t(s_i) - \overline{Q_t(s)})^2}{N}}$ <p>where $Q_t(s_i)$ stands for the response time of service s connection at the ith time and $\overline{Q_t(s)}$ is the average response time of service. $Q_s(s)$ is applied to evaluate the performance of stability of a cloud service.</p>
Additional information	Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non-functional

Measurement	Resilience
Level	Assurance
Definition	<p>Is the ability of a system exposed to hazards to resist and recover in a timely and efficient manner, including the preservation and restoration of its essential basic structures and functions.</p> <p>A resilient system is one that shows reduced failure probabilities, reduced consequences from failures, and reduced time to recovery. It is supported in cloud architectures by employing redundant connections and by duplicating resources or information. Resilience by itself is characterized by several metrics, such as availability and security. Mechanisms aiming to maximize it must consider these criteria as well as common network metrics, i.e., path capacity, one-way delay and packet loss.</p>
Way to perform calculation	Not defined by the authors
Additional information	Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non-functional

TQACA - Empathy

Measurement	Empathy
Level	Root
Definition	Customer experiences based on the expectations. Cloud service providers should try to cover the various demands from their customers

	and should provide courteous service to their customers. Therefore we define Empathy as the degree to which the cloud service adapts the service to the needs of individual customers.
Way to perform calculation	It depends on the children nodes
Additional information	Type: Qualitative Unit: Descriptive Scope: User Category: Objective Requirement: Non-functional

Measurement	Courtesy
Level	Empathy
Definition	Degree to which a cloud service is provided in a polite, respectful and friendly way
Way to perform calculation	Degrees of formalization of the employee's language and attitude (DoF-ATT) s = 3 when service provider has established sound specification on polite service and employee can fulfill the specification well; s = 2 when service provider has established specification on polite service, but employee cannot fulfill the specification very well; s = 1 when service provider has not established specification on polite service and employee can politely provide service comparatively; s = 0 when there is no specification on polite service and employee have a bad attitude. $d = s / 3$, d = DoF-ATT, $0 \leq d \leq 1$ The closer the value of "d" is to 1, the better the DoF-ATT.
Additional information	Type: Quantitative Unit: Scale Scope: User Category: Subjective Requirement: Non-functional

Measurement	Initiative
Level	Empathy
Definition	Degree to which a cloud service recognizes customers' requirements and suggests changes to meet their needs
Way to perform calculation	Not defined by the authors

Additional information	Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non-functional
------------------------	--

Measurement Level	Measurability Empathy
Definition	The ability that usage of a cloud service can be monitored, controlled, reported and billed.
Way to perform calculation	Not defined by the authors
Additional information	Type: Quantitative Unit: Numeric Scope: User Category: Objective Requirement: Non-functional

Measurement Level	Satisfaction Empathy
Definition	A judgment that a service provided a pleasurable level of consumption related fulfillment.
Way to perform calculation	These studies classify product/service attributes into two categories: (1) basic, lower order, or hygiene needs and (2) growth, higher order, or motivator needs.
Additional information	Type: Qualitative Unit: Scale Scope: User Category: Subjective Requirement: Non-functional

Measurement Level	Self-service Empathy
Definition	Degree to which a customer can unilaterally provision and manage the cloud resources as needed
Way to perform calculation	Not defined by the authors
Additional information	Type: Quantitative Unit: Numeric Scope: User Category: Objective

	Requirement: Non-functional
--	-----------------------------

TQACA - Financial

Measurement	Financial
Level	Root
Definition	<p>The amount of money spent on the service by the client.</p> <p>The amount of money spent on the service by the client. This is clearly one of the vital attributes for IT and the business. Cost tends to be the single most quantifiable metric today, but it is important to express cost in the characteristics, which are relevant to a particular business organization.</p>
Way to perform calculation	It depends on the children nodes
Additional information	Type: Quantitative Unit: Currency Scope: User Category: Objective Requirement: Non-functional

Measurement	Billing process
Level	Financial
Definition	The level of integration that is available between the client and cloud service provider's billing systems and the predictability of periodic bills
Way to perform calculation	Not defined by the authors
Additional information	Type: Quantitative Unit: Numeric Scope: User Category: Objective Requirement: Non-functional

Measurement	Cost
Level	Financial
Definition	The client's cost to consume a service over time. This includes cost of transition of the service along with recurring costs. (e.g. monthly access fees) and usage-based costs.

	<p>Refers to the cost imposed by the cloud computing provider for the development and implementation of the cloud computing technology and service in an organization. This usually includes both the acquisition and on-going costs of the cloud computing technology.</p>
Way to perform calculation	<p>$Q_p(s)$ to denote the price of service s.</p> <p><i>Another formula:</i></p> <p>The cost of service s for a given configuration d, depends on the cost of the used VM instances in the $\#vmg$ VM groups:</p> $c_{s,d} = \sum_{i=1}^{\#vmg} c_{inst}(v_d) \times inst_i(v_d) + c_{use}(serv_s)$ <p>In this equation, for each VM group i, $c_{inst}(v_d)$ returns the cost of a particular instance type and instance size in configuration d. $inst_i(v_d)$ returns the number of instances used for VM groups i in configuration d. Cost is then the sum of the costs for all VM instances in configuration d.</p> <p>For PaaS and SaaS, cost usually depends on several elements in the service specification, $serv_s$, e.g. work, load or quality thresholds. This is specified by the term $c_{use}(serv_s)$. Cost can also depend on internal properties of the service, for example, number of payment operations required; factors which depend on $c_{use}(serv_s)$ in more or less complex ways.</p>
Additional information	<p>Type: Quantitative Unit: Numeric Scope: User Category: Subjective Requirement: Non-functional</p>
Measurement	Data Transfer Cost
Level	Financial » Cost
Definition	The cost to transfer data in or out of data center (\$/Gb)
Way to perform calculation	Not defined by the authors
Additional information	<p>Type: Quantitative Unit: Numeric Scope: User Category: Objective Requirement: Non-functional</p>

Measurement	Liquidated damages
Level	Financial » Cost
Definition	The available compensation for the cloud service requester, is released by the cloud service provider for cloud requester.
Way to perform calculation	Not defined by the authors
Additional information	Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non-functional

Measurement	Storage Cost
Level	Financial » Cost
Definition	The cost of secondary storage (\$/Gb/Month)
Way to perform calculation	N/A (Not Applicable)
Additional information	Type: Quantitative Unit: Numeric Scope: User Category: Objective Requirement: Non-functional

Measurement	VM Cost
Level	Financial » Cost
Definition	The cost to deploy a VM (\$/hr or \$/month).
Way to perform calculation	N/A (Not Applicable)
Additional information	Type: Quantitative Unit: Numeric Scope: User Category: Objective Requirement: Non-functional

TQACA - Performance

Measurement	Performance
Level	Root
Definition	This category covers the features and functions of the provided services.

	<p>This category covers the features and functions of the provided services. There are many different solutions offered by cloud providers addressing the IT needs of different organizations.</p> <p>Each solution has different performance in terms of functionality, service response time, accuracy and so on.</p>
Way to perform calculation	$Rate(p) = \frac{Bytes(p)}{Page(t)}$ <p>Where Rate(p) is page download rate, Bytes(p) is the number of bytes of the requested page, Page(t) is the download time of the requested page. Generally, performance formulas reflect the specific domain they refer to.</p>
Additional information	<p>Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Functional</p>

Measurement	Accuracy
Level	Performance
Definition	The extent to which a service adheres to its requirements
Way to perform calculation	<p>AoS = 1 - $\frac{\text{number of failed requests}}{\text{total number of requests}}$</p> <p>where the denominator is a total number of requests from users' and the numerator is a number of failed requests.</p>
Additional information	<p>Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non-functional</p>

Measurement	Capacity
Level	Performance
Definition	Capacity means maximum amount of resources that a service provider can provide at peak times
Way to perform calculation	<p>Let $cpu_u, mem_u, sto_u, BW_u$ be required values of CPU, memory, storage and bandwidth respectively by user and $cpu_{sp}, mem_{sp}, sto_{sp}, BW_{sp}$ be vectors of offered service providers value of CPU, memory, storage and bandwidth. Collectively, it can be written as follows.</p> <p>$cpu = \{cpu_u, cpu_{sp}\}$ $mem = \{mem_u, mem_{sp}\}$</p>

	<p> $sto = \{sto_u, sto_{sp}\}$ $BW = \{BW_u, BW_{sp}\}$ </p> <p>Normalized value of cpu, mem, sto, BW are</p> <p> $cpu_N = cpu / \max\{cpu\} = \{cpu_{N,u}, cpu_{N,sp}\}$ $mem_N = mem / \max\{mem\} = \{mem_{N,u}, mem_{N,sp}\}$ $sto_N = sto / \max\{sto\} = \{sto_{N,u}, sto_{N,sp}\}$ $BW_N = BW / \max\{BW\} = \{BW_{N,u}, BW_{N,sp}\}$ </p> <p>To differentiate service providers clearly, if normalized offered value of a service provider is less than required normalized value by user, normalized offered value is made zero. It can be written as:</p> <p> $cpu_{N,sp}(i) = \{cpu_{N,sp}(i), cpu_{N,sp}(i) \geq cpu_{N,u}\}$ $0, \text{ otherwise}\}$ $mem_{N,sp}(i) = \{mem_{N,sp}(i), mem_{N,sp}(i) \geq mem_{N,u}\}$ $0, \text{ otherwise}\}$ $sto_{N,sp}(i) = \{sto_{N,sp}(i), sto_{N,sp}(i) \geq sto_{N,u}\}$ $0, \text{ otherwise}\}$ $BW_{N,sp}(i) = \{BW_{N,sp}(i), BW_{N,sp}(i) \geq BW_{N,u}\}$ $0, \text{ otherwise}\}$ </p> <p> $F = [cpu_{N,sp}', mem_{N,sp}', sto_{N,sp}', BW_{N,sp}']$ </p> <p>Capacity of service provider <i>i</i> is depicted as follows.</p> <p> $capacity_i = \text{summation } j=1 \text{ to } 4 \{F_{ij} * W_j\};$ </p> <p>where w_1, w_2, w_3, w_4 are weights of capacity of CPU, memory, storage and network respectively for an application.</p>
Additional information	<p>Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non-functional</p>
Measurement	CPU capacity metric
Level	Performance » Capacity
Definition	Measured by multiplication of number of CPU and frequency of CPU
Way to perform calculation	Let $cpu_u, mem_u, sto_u, BW_u$ be required values of CPU, memory, storage and bandwidth respectively by user and $cpu_{sp}, mem_{sp}, sto_{sp}, BW_{sp}$ be vectors of offered service providers value of CPU, memory, storage and bandwidth. Collectively, it can be written as follows.

	<p>$cpu = \{cpu_u, cpu_{sp}\}$</p> <p>Normalized value of cpu, mem, sto, BW are</p> <p>$cpu_N = cpu / \max\{cpu\} = \{cpu_{N,u}, cpu_{N,sp}\}$</p>
Additional information	<p>Type: Quantitative</p> <p>Unit: Numeric</p> <p>Scope: Provider</p> <p>Category: Objective</p> <p>Requirement: Non-functional</p>

Measurement	CPU speed
Level	Performance » Capacity » CPU capacity metric
Definition	Maximum number of instructions executed by a single core in unit time (MIPS).
Way to perform calculation	Maximum number of instructions executed by a single core in unit time (MIPS).
Additional information	<p>Type: Quantitative</p> <p>Unit: Numeric</p> <p>Scope: Provider</p> <p>Category: Objective</p> <p>Requirement: Non-functional</p>

Measurement	CPU occupancy rate
Level	Performance » Capacity
Definition	It refers to the proportion of service instances require computing capability accounting the total available computing capability of service node.
Way to perform calculation	$Cor = C_{si} / C_s$ <p>Where C_i is the computing capability of i-th service required in Service Portfolio. C_s is available computing power of i-th service in the service node.</p>
Additional information	<p>Type: Quantitative</p> <p>Unit: Numeric</p> <p>Scope: Provider</p> <p>Category: Objective</p> <p>Requirement: Non-functional</p>

Measurement	Memory capacity metric
Level	Performance » Capacity
Definition	Measured in GB

Way to perform calculation	<p>Let cpu_u, mem_u, sto_u, BW_u be required values of CPU, memory, storage and bandwidth respectively by user and cpu_{sp}, mem_{sp}, sto_{sp}, BW_{sp} be vectors of offered service providers value of CPU, memory, storage and bandwidth. Collectively, it can be written as follows.</p> $mem = \{mem_u, mem_{sp}\}$ <p>Normalized value of cpu, mem, sto, BW are</p> $mem_N = mem / \max\{mem\} = \{mem_{N,u}, mem_{N,sp}\}$
Additional information	<p>Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non-functional</p>

Measurement	Memory Read
Level	Performance » Capacity » Memory capacity metric
Definition	Maximum amount of data transferred from memory in unit time (Mb/s)
Way to perform calculation	Not defined by the authors
Additional information	<p>Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non-functional</p>

Measurement	Memory Write
Level	Performance » Capacity » Memory capacity metric
Definition	Maximum amount of data written to memory in unit time (Mb/s).
Way to perform calculation	Not defined by the authors
Additional information	<p>Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non-functional</p>

Measurement	Network capacity metric
Level	Performance » Capacity
Definition	Measured by bandwidth

Way to perform calculation	<p>Let cpu_u, mem_u, sto_u, BW_u be required values of CPU, memory, storage and bandwidth respectively by user and cpu_{sp}, mem_{sp}, sto_{sp}, BW_{sp} be vectors of offered service providers value of CPU, memory, storage and bandwidth. Collectively, it can be written as follows.</p> <p>$BW = \{BW, BW\}$</p> <p>Normalized value of cpu, mem, sto, BW are</p> <p>$BW_N = BW / \max\{BW\} = \{BW_{N,u}, BW_{sp}\}$</p>
Additional information	<p>Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non-functional</p>

Measurement	Inbound
Level	Performance » Capacity » Network capacity metric
Definition	Maximum amount of data transferred into VM in unit time (Gbit/s).
Way to perform calculation	Not defined by the authors
Additional information	<p>Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non-functional</p>

Measurement	Outbound
Level	Performance » Capacity » Network capacity metric
Definition	Maximum amount of data transferred out of VM in unit time (Gbit/s).
Way to perform calculation	Not defined by the authors
Additional information	<p>Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non-functional</p>

Measurement	Task integrity
Level	Performance » Capacity » Network capacity metric
Definition	Integrity, here we are referring to the integrity of the tasks. The worse the

	situation of network is, the easier to lose the tasks are.
Way to perform calculation	<p>It uses the task loss rate (referred to TL) to represent the integrity, is defined as:</p> $TL_{Pr}^n = \frac{N_r^{loss}}{N_r^{tot}}$ <p>TL_{Pr}^n represents the task loss rate of the n-th transaction for the cloud providers r. N_r^{loss} represents the number of lost tasks in the n-th transaction, N_r^{tot} represents the total number of tasks in the n-th transaction for the cloud providers r.</p>
Additional information	<p>Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non-functional</p>

Measurement	Storage capacity metric
Level	Performance » Capacity
Definition	Measured in GB
Way to perform calculation	<p>Let cpu_u, mem_u, sto_u, BW_u be required values of CPU, memory, storage and bandwidth respectively by user and cpu_{sp}, mem_{sp}, sto_{sp}, BW_{sp} be vectors of offered service providers value of CPU, memory, storage and bandwidth. Collectively, it can be written as follows.</p> $sto = \{sto_u, sto_{sp}\}$ <p>Normalized value of cpu, mem, sto, BW are</p> $sto_N = sto / \max\{sto\} = \{sto_{N,u}, sto_{N,sp}\}$
Additional information	<p>Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non-functional</p>

Measurement	Disk Read
Level	Performance » Capacity » Storage capacity metric
Definition	Maximum amount of data transferred from secondary storage in unit time

	(Mb/s)
Way to perform calculation	Not defined by the authors
Additional information	Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non-functional

Measurement	Disk Write
Level	Performance » Capacity » Storage capacity metric
Definition	Maximum amount of data written to secondary storage in unit time (Mb/s).
Way to perform calculation	Not defined by the authors
Additional information	Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non-functional

Measurement	Data transfer
Level	Performance
Definition	Data transfer rate measures how fast the service is provided
Way to perform calculation	Not defined by the authors
Additional information	Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non-functional

Measurement	Functionality
Level	Performance
Definition	The specific features provided by a service
Way to perform calculation	We choose 4 submetrics such as suitability (SU), accuracy (AC), interoperability (IO), and security (SEC).
Additional information	Type: Quantitative Unit: Numeric Scope: Application Category: Objective Requirement: Non-functional

Measurement	Interoperability
Level	Performance
Definition	The ability of a service to easily interact with the others (from the same cloud service provider and from other cloud service providers)
Way to perform calculation	For calculating the value of this metric, we define a term IO denoting interoperability as in the following equation: IO = 1 - $\frac{\text{No. of failures when data exchanges}}{\text{No. of total data exchanges}}$
Additional information	Type: Quantitative Unit: Numeric Scope: Developer Category: Objective Requirement: Non-functional

Measurement	Specific Service
Level	Performance
Definition	If the organization offers a high quality service, it is generally recognized by that service, therefore the CSP will most likely be selected.
Way to perform calculation	Not defined by the authors
Additional information	Type: Qualitative Unit: Descriptive Scope: Provider Category: Objective Requirement: Non-functional

Measurement	Suitability
Level	Performance
Definition	Reflects on the degree to which the organization's requirements for specific uses or applications are met by the cloud computing provider. Here, the cloud computing provider must satisfy both critical and noncritical requirements of the organization.
Way to perform calculation	For calculating the value of this metric, we define a term SU denoting suitability as in the following equation: SU = $\frac{\text{Number of missing functions}}{\text{Number of required functions}}$
Additional information	Type: Quantitative Unit: Numeric Scope: Provider

	Category: Objective Requirement: Non-functional
Measurement	Dedicated hardware
Level	Performance » Suitability
Definition	Measuring service readiness to cope with the availability issues that result from sharing resources. Considerations are: <ol style="list-style-type: none"> 1. Does the CSP offer dedicated machines to a single consumer? 2. Does the CSP apply the same security countermeasures that apply to VMs on dedicated machines?
Way to perform calculation	Not defined by the authors
Additional information	Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non-functional

Measurement	Throughput and Efficiency
Level	Performance
Definition	Throughput means number of tasks completed per unit time by Cloud service provider. An application is constituted as collection of tasks (n). It may be an embarrassingly parallel application in which all tasks are independent and can be executed in any order. On the other side, all tasks may be dependent on each other. An application may belong to high performance computing, high-throughput computing or many-task computing. So throughput of an application not only depends on service provider but also on application itself.
Way to perform calculation	n , number of tasks T , task execution time of an application To , the overhead time of cloud data center So, throughput (Thrapp) of service provider for an application is given by: Thrapp = n / T + To Efficiency (eapp) of a service provider is given by: eapp = T / T - To
Additional information	Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non-functional

Measurement	Marginal costs
Level	Performance » Throughput and Efficiency
Definition	(MC) We derive the marginal costs for a specified workload delta from EMprov. Marginal costs are the operation costs to serve one additional workload unit, thus, measuring the efficiency of a cloud computing system.
Way to perform calculation	For example, the operation costs to serve 20% additional requests per second (factor 1.2) can be \$1.00 for system SA, i.e., MCSA(1.2) = \$1.00.
Additional information	Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non-functional

Measurement	Resource Provisioning Efficiency
Level	Performance » Throughput and Efficiency
Definition	(RPE) We define resource provisioning efficiency (RPE) as a metric to measure a system's efficiency in a specified workload delta based on EMclose. That is, the metric measures the mismatch between actual resource utilization and resource demand while the workload is changing. We measure this mismatch in percentage, i.e., as a real number. A perfectly efficient system will adapt its resource demand exactly to the resource demand at all times.
Way to perform calculation	For example, if the workload increases with factor 1.2 the system will provision exactly that amount of additional resources required to cope with this additional workload, i.e., RPESA(1.2 req/s) = 0.
Additional information	Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non-functional

Measurement	Utilization
Level	Performance
Definition	The expected data center utilization U can be computed as the ratio between the number of physical resources used at steady state and the total number N of physical resources.
Way to perform calculation	Defining the following reward function: $r^u = P_{run}^\#$ and considering the presence of virtual resources, it can be computed as:

	$U = \min\{1, E[r_X^u]/N\}$
Additional information	Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non-functional

TQACA - Security and privacy

Measurement	Security and privacy
Level	Root
Definition	<p>This category includes attributes that indicate the effectiveness of a cloud service providers controls on access to services, service data, and the physical facilities from which services are provided.</p> <p>The degree to which a cloud service protects both customer assets and access to their information and data so that persons or other products or systems have the degree of data access appropriate to their types and levels of authorization.</p>
Way to perform calculation	<p>$\theta = 1 - F_T(t)$ where θ represents security and $F_T(t)$ denotes a cumulative distribution function of a random variable T indicating the time until the first security breach occurs, measured in unit time.</p> <p>For simplicity, we assume that the security issues occur at random and uniformly in cloud services throughout a time interval. That is, there is no clustering of security issues. The occurrence of security issues, then, can be modeled as a Poisson process with mean $\lambda > 0$. Let a random variable T denote the time from the start of an operational period until the first security breach occurs. T now follows an exponential distribution with parameter λ, whose cumulative distribution function is described as:</p> $F_T(t) = 1 - e^{-\lambda t} \quad t \geq 0$ <p>Alternatively, its probability density function is determined, by differentiation, as</p> $f_T(t) = \lambda e^{-\lambda t} \quad t \geq 0$
Additional information	Type: Quantitative Unit: Numeric Scope: User Category: Objective Requirement: Functional

Measurement	Access Control
Level	Security and privacy
Definition	<p>An access control system includes components and methods to specify access control policies for legitimate users.</p> <p>Considerations include: (1) Does the CSP provide Separation of Duties and Administrator Access?, (2) Does the CSP allow full control on the ACLs?</p>
Way to perform calculation	<p>It involves six factors, namely:</p> <p>Controllability (F1) = { f1, f2, f9, f10, f12, f18}</p> <p>Service article f1 Service review f2 ID identification f9 Access control f10 Data encryption f12 Software and hardware fault monitoring/recovering f18</p>
Additional information	<p>Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non-functional</p>

Measurement	Attribute based Encryption
Level	Security and privacy » Access Control
Definition	<p>The uploaded data is encrypted using Attribute based Encryption (ABE) that defines access policy on attributes related to the data. Hence only authorized users with matching attributes can decrypt and access the data.</p> <p>Identifies data protection on its way from consumer to cloud and in transit against unauthorized use by scrambling the contents so that it can be read only by someone who has the encryption key to unscramble it.</p> <p>Considerations include: Is data transferred to and from the cloud encrypted by default? Is the encryption attribute FIPS 140-2 compliant?</p>
Way to perform calculation	Not defined by the authors
Additional information	<p>Type: Quantitative Unit: Numeric Scope: Provider Category: Objective</p>

	Requirement: Non-functional
Measurement	Authentication and Authorization
Level	Security and privacy » Access Control
Definition	Authentication and Authorization (A&A) will be a major challenge in NGI. A&A will be at two major levels – at the user level within the network and at the application level at the computing platform end. In the NGI the user will seamlessly roam between networks. These networks can be a wired network or wireless network – WiFi or 3G or even WiMAX networks. Along with mobile IP, the network will provide seamless A&A, handoff, and roaming between homogenous and heterogenous networks.
Way to perform calculation	Not defined by the authors
Additional information	Type: Qualitative Unit: Descriptive Scope: Provider Category: Objective Requirement: Functional
Measurement	Identity Management and Authentication
Level	Security and privacy » Access Control
Definition	Identifies the service robustness against unauthorized access to the CC components (e.g. control panel, VMs, GUI, database, etc.) by verifying login credentials. Considerations include: (1) Is multi-factor authentication (MFA) supported?, (2) Is the authentication system synchronized with the organization “active directory”?
Way to perform calculation	Not defined by the authors
Additional information	Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Functional
Measurement	Revocation
Level	Security and privacy » Access Control
Definition	It seeks ways to invalidate information used by authentication and authorization dynamically. So, the user or entity, when is transferring credentials, is performing authentication or is releasing attributes among different cloud providers, can be assured immediate suspension of the request.

Way to perform calculation	Not defined by the authors
Additional information	Type: Qualitative Unit: Descriptive Scope: Provider Category: Objective Requirement: Functional
Measurement Level	Backup Security and privacy
Definition	The ability to archive code and data of an application at specific times. Measures service readiness to respond to failure, loss, or damage in clouds (e.g. Virtual Machines (VMs), data, etc.) by making duplicate(s) of originals. Considerations include: Is backup/restore bandwidth unlimited? Is the backup service automated?
Way to perform calculation	Not defined by the authors
Additional information	Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non - functional
Measurement Level	Confidentiality Security and privacy
Definition	The ability that a cloud service ensures that data are accessible only to those authorized to have access. Policies and processes in use by the cloud service provider to ensure that only the personnel granted appropriate privileges can make use of or modify data / work products
Way to perform calculation	Not defined by the authors
Additional information	Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non - functional

Measurement	Data Security
Level	Security and privacy
Definition	Score of security mechanisms that protect users' data.
Way to perform calculation	Not defined by the authors
Additional information	Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non - functional

Measurement	Data accuracy
Level	Security and privacy » Data Security
Definition	It refers to the degree of closeness between the observed result and the true value or value that is accepted as true. Therefore, we can know this quality parameter is typically used to measure the collected sensor data by comparing the multiple sources.
Way to perform calculation	Not defined by the authors
Additional information	Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non - functional

Measurement	Data Availability
Level	Security and privacy » Data Security
Definition	Data Availability (DA) is related to the frequent range of access to a specified file and has a direct effect to IC. In the proposed model, the value of DA has been determined according to the total number of downloading and uploading of file F.
Way to perform calculation	The total numbers of downloading and re-uploading specific file are directly proportional to the value of DA. In fact, this value is calculated according to the number of digits in NU and NA. For instance, if total number of downloading file F is 36467, the total number of uploading file F is 2523, and the maximum number of downloads is 1000000, i, j, and k will be 7, 5.3, and 4.2 respectively. Thus, $E(ND) = 1.7$ and $E(NU) = 2.2$ and the value of data availability will be 3.9. If the value of data availability reaches 5, protection of file will be in the lowest level.
Additional information	Type: Quantitative Unit: Numeric Scope: Provider Category: Objective

	Requirement: Non - functional
Measurement	Data Confidentiality
Level	Security and privacy » Data Security
Definition	Client restrictions on use and sharing of client data are enforced by the cloud service provider. Any failures of these protections are promptly detected and reported to the client
Way to perform calculation	Thus, four main parameters have been defined for determination of DC. The value of DC is calculated according to direct proportional of UA and UE, and inverse proportional of UR. In fact, determining the value of DC due to the number of accessed or revoked users enhance the reliability of IC value that is explained in subsequent sections.
Additional information	Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non - functional

Measurement	Data consistency
Level	Security and privacy » Data Security
Definition	Data consistency refers to data collection methods, schedules, and locations. It is much helpful to evaluate the consistency of the big data sets in abundant and different angles.
Way to perform calculation	<p>We propose the following metrics to measure degrees of consistency: Staleness, Violations of MRC, Violations of MWC, Violations of RYWC.</p> <p>Staleness: This is the time window between the last read of version n and the start of the write of version n+1 for most storage systems. If the system guarantees transaction isolation, this value decreases by the write latency for version n + 1. Given a distribution of these inconsistency windows, one can also calculate a curve describing the probability of stale reads over time which is just a different representation of the same data.</p> <p>Violations of MRC: This the likelihood of reading a version n followed by reading an older version some time later.</p> <p>Violations of MWC: The likelihood of two consecutive writes by the same client being serialized in reverse chronological order.</p> <p>Violations of RYWC: The likelihood of a client writing a version n and the same client reading an older version some time later.</p>
Additional information	Type: Quantitative Unit: Numeric

	Scope: Provider Category: Objective Requirement: Non - functional
--	---

Measurement	Data correctness
Level	Security and privacy » Data Security
Definition	This data quality parameter is much helpful to evaluate the correctness of big data sets in term of data types, formats, and so on.
Way to perform calculation	Not defined by the authors
Additional information	Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non - functional

Measurement	Data geographic /political
Level	Security and privacy » Data Security
Definition	The client's constraints on service location based on geographic or political risk. Identifies the CSP readiness to protect cloud service data centers (i.e. compute, storage, etc.) against the physical security risks. Considerations include: (1) Does the CSP hide the exact locations of the data centers from the public?, (2) Can consumers choose the geographical location of where to store their data?
Way to perform calculation	Not defined by the authors
Additional information	Type: Quantitative Unit: Numeric Scope: User Category: Objective Requirement: Non - functional

Measurement	Data Integrity
Level	Security and privacy » Data Security
Definition	The ability that a cloud service prevents unauthorized access to, or modification of, the customer's information or data. Keeping the data that is created, used, and stored in its correct form so that clients may be confident that it is accurate and valid. Here we are referring to the integrity of the tasks. Poor thing in the network, the tasks are often lost, which usually occurred in the poor

	network.
Way to perform calculation	<p>If C_k is the number of works completed by R_k source and D_k is the number of data integrity works provided by R_k in T time limit so data integrity is obtained according to equation:</p> $\text{Data Integrity (DI)} R_k = \frac{D_k}{C_k}$ <p>The task loss rate (referred to TL) represents as below.</p> $TL_n = N^n_{\text{loss}} / N^n_{\text{tot}}$
Additional information	<p>Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non - functional</p>

Measurement	Data Sanitization
Level	Security and privacy » Data Security
Definition	<p>Identifies the security measures that CSP take to deliberately, permanently, and irreversibly remove or destroy data (in storage media, documents, etc.) so that it can never be recovered or retrieved. Consideration are: (1) Is the data destroyed securely when the service is terminated?, (2) Does the CSP sanitize the storage media before being reused?</p>
Way to perform calculation	Not defined by the authors
Additional information	<p>Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non - functional</p>

Measurement	Network security support
Level	Security and privacy
Definition	<p>Identifies the CSP's effectiveness in preventing unauthorized access, misuse, modification, or denial of cloud physical and virtual networks. Considerations include: (1) Does the CSP allow consumers to build geographically dispersed networks?, (2) Does the CSP provide countermeasures for all network attacks (e.g. MITM, DDoS, etc.)?</p>
Way to perform calculation	Not defined by the authors
Additional information	<p>Type: Quantitative Unit: Numeric</p>

	Scope: Provider Category: Objective Requirement: Non - functional
--	---

Measurement	Obligations
Level	Security and privacy
Definition	Actions that must be performed when certain events occur. When the event is an authorization decision, the obligations are actions that should follow this decision. For example, if a particular provider does not comply with data usage policies to a user regarding legislation, the user will be exposed to unauthorized access.
Way to perform calculation	Not defined by the authors
Additional information	Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non - functional

Measurement	Physical and environmental security
Level	Security and privacy
Definition	Policies and processes in use by the cloud service provider to protect the provider facilities from unauthorized physical access, damage or interference. Measures the CSP's ability to provide knowledge to all cloud stakeholders to increase protection of the physical, and informational assets in clouds. Considerations include: (1) Does the CSP provide recommendations for better security?, (2) Does the CSP warn consumers of possible vulnerabilities?
Way to perform calculation	Not defined by the authors
Additional information	Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non - functional

Measurement	Proactive threat and vulnerability management
Level	Security and privacy
Definition	Mechanism in place to ensure that the service is protected against know recurring threats as well as new evolving vulnerabilities.

	Measure the robustness of the process of ensuring the VM manager (VMM) is secure. This includes its computations, networking, and storage. Considerations include: (1) Does the CSP prevent deployment of insecure or tampered VMs?, (2) Can the CSP identify and defend against VM side-channel attack?
Way to perform calculation	Not defined by the authors
Additional information	Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non - functional

Measurement	Risks
Level	Security and privacy
Definition	The user should have the autonomy, knowing the risks, to choose a particular server or service and also to have trust to release his data. Assesses the CSP goodness in identifying, assessing, and prioritizing, monitoring, and controlling risks. Considerations include: (1) Does the CSP have a risk management plan for the offered services?, (2) Does the CSP allow for certified external auditors' involvement in all risk management activities?
Way to perform calculation	Not defined by the authors
Additional information	Type: Qualitative Unit: Descriptive Scope: User Category: Subjective Requirement: Non - functional

Measurement	Safety
Level	Security and privacy
Definition	The safety of service s, denoted $Q_{sa}(s)$, is influenced by the security technologies deployed in the mobile cloud service and its provision.
Way to perform calculation	We divide $Q_{sa}(s)$ into three levels. One example setting is if the cloud service supports Token, Time Stamp, Signature, and Encrypted for Web Services Security (WSS), $Q_{sa}(s)$ is set as Level 1, else if the cloud service support Secure Sockets Layer (SSL), $Q_{sa}(s)$ falls into Level 2, otherwise $Q_{sa}(s)$ is set as Level 3.
Additional information	Type: Quantitative Unit: Numeric Scope: Provider

	Category: Objective Requirement: Non - functional
Measurement	Security management
Level	Security and privacy
Definition	The capabilities of cloud service providers to ensure application, data, and infrastructure security based on the security requirements of the client. Considerations include: (1) Does the CSP encourage secure protocols like SSL, IPsec? (2) Does the CSP make best effort to increase consumers' S&P awareness?
Way to perform calculation	Not defined by the authors
Additional information	Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non - functional

Measurement	Traceability
Level	Security and privacy
Definition	The ability that the activities in the process of service by the cloud service provider are recorded and can be traced.
Way to perform calculation	Not defined by the authors
Additional information	Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non - functional

TQACA - Usability

Measurement	Usability
Level	Root
Definition	The degree to which a cloud service can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use. The ease with which a service can be used.

	Usability (USAB) describes how easy, efficient, and enjoyable the interface to a cloud service is. Assesses the ease of invocation if the functionality of a cloud service is exposed as Application Programming Interface (API).
Way to perform calculation	$T_U = \frac{\text{Number of successful responses}}{\text{Total number of requests}}$ <p>Here, T_U represents the mathematical trustworthiness value of usability while calculating overall trustworthiness value of cloud service.</p>
Additional information	Type: Qualitative Unit: Descriptive Scope: User Category: Subjective Requirement: Non-functional

Measurement	Accessibility
Level	Usability
Definition	The ease with which information can be accessed or extracted from the system.
Way to perform calculation	<p>(Lee et al., 2003). Mathijssen (2005) denoted accessibility as follows:</p> $QoS_{\text{Accessibility}} = \frac{\text{The number of user's requests that succeeded}}{\text{Total requests done by the user}}$ <p>Similarly, Yu et al. (2007) calculate accessibility as a ratio of the number of successful acknowledgements received to the total number of requests sent, as shown below:</p> $QoS_{\text{Accessibility}} = \frac{\text{Number of acknowledgements received}}{\text{Total number of requests sent}}$
Additional information	Type: Quantitative Unit: Numeric Scope: Provider Category: Subjective Requirement: Functional

Measurement	Appropriateness Recognizability
Level	Usability
Definition	Degree to which users can recognize whether a cloud service is appropriate for their needs.
Way to perform calculation	Not defined by the authors
Additional	Type: Quantitative

information	Unit: Scala Scope: User Category: Subjective Requirement: Non-functional
-------------	---

Measurement	Changeability
Level	Usability
Definition	It can be formulated as follows: $\text{Changeability(MTTC)} = \frac{\sum (\text{Time to analyze the change in workload} + \text{Time to modify the change in workload} + \text{Time to test the change in workload} + \text{time to distribute the change in workload})}{\text{No. of change requests in workload}}$
Way to perform calculation	It can be formulated as follows: $\text{Changeability(MTTC)} = \frac{\sum (\text{Time to analyze the change in workload} + \text{Time to modify the change in workload} + \text{Time to test the change in workload} + \text{time to distribute the change in workload})}{\text{No. of change requests in workload}}$
Additional information	Type: Quantitative Unit: Numeric Scope: Developer Category: Objective Requirement: Non-functional

Measurement	Client Interface
Level	Usability
Definition	Degree to which the interface of the cloud service enables pleasing and satisfying interaction for the customer
Way to perform calculation	Not defined by the authors
Additional information	Type: Qualitative Unit: Descriptive Scope: User Category: Subjective Requirement: Non-functional

Measurement	Documentation
Level	Usability
Definition	Amount of documentation in WSDL file.
Way to perform calculation	Not defined by the authors
Additional information	Type: Quantitative Unit: Numeric Scope: Developer Category: Objective Requirement: Functional

Measurement	Free Trial
Level	Usability
Definition	Some Providers provide free trial to test their services. It is very beneficial for users. User can test services before deployment.
Way to perform calculation	Not defined by the authors
Additional information	Type: Qualitative Unit: Boolean Scope: User Category: Objective Requirement: Non-functional

Measurement	Group Formation
Level	Usability
Definition	The process needed to identify group members and make a group.
Way to perform calculation	Not defined by the authors
Additional information	Type: Quantitative Unit: Numeric Scope: User Category: Objective Requirement: Non-functional

Measurement	Learnability
Level	Usability
Definition	Degree to which a cloud service can be learned by users to achieve a specified level of effectiveness, efficiency, and freedom from risk within a specified amount of time and context of use. The effort required of users to learn to use the service
Way to perform calculation	Not defined by the authors
Additional information	Type: Quantitative Unit: Numeric Scope: Developer Category: Objective Requirement: Non-functional

Measurement	API
Level	Usability » Learnability

Definition	Whether provider is providing an API to interact with server or not is also an important parameter.
Way to perform calculation	The formula is a range of values: S = {None, Average, Extensive}
Additional information	Type: Qualitative Unit: Scala Scope: Developer Category: Objective Requirement: Non-functional

Measurement	Operability
Level	Usability
Definition	Degree to which a cloud service has attributes that make it easy to operate and control. The ability of a service to be easily operated by users
Way to perform calculation	Not defined by the authors
Additional information	Type: Qualitative Unit: Scala Scope: Developer Category: Subjective Requirement: Non-functional

Measurement	Revision
Level	Usability
Definition	Tracking revision history and the individual responsible for a revision
Way to perform calculation	Not defined by the authors
Additional information	Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Functional

Measurement	Share
Level	Usability
Definition	Allowing an application to be shared by other people.
Way to perform calculation	Not defined by the authors
Additional	Type: Quantitative

information	Unit: Numeric Scope: User Category: Objective Requirement: Non-functional
-------------	--

Measurement	Synchronization
Level	Usability
Definition	Adjusting phases of different systems so systems oscillate together.
Way to perform calculation	Not defined by the authors
Additional information	Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Functional

Measurement	Tangibility
Level	Usability
Definition	Can be defined as the degree to which the tangible aspects of the cloud service effectively communicate and support the service.
Way to perform calculation	Not defined by the authors
Additional information	Type: Qualitative Unit: Descriptive Scope: Provider Category: Subjective Requirement: Non-functional

Measurement	Professionalism
Level	Usability » Tangibility
Definition	Degree to which the cloud service is professional and based on appropriate skill and expertise.
Way to perform calculation	Not defined by the authors
Additional information	Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non-functional

Measurement	Visibility
-------------	------------

Level	Usability » Tangibility
Definition	Degree to which potential customers are aware of the cloud service.
Way to perform calculation	Not defined by the authors
Additional information	Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non-functional

Measurement	Transparency
Level	Usability
Definition	The extent to which users are able to determine when changes in a feature or component of service occur and whether these changes impact usability. Transparency is a property of a system that it is capable of “giving account” of, or providing visibility of, how it conforms to its governing rules and commitments.
Way to perform calculation	Not defined by the authors
Additional information	Type: Quantitative Unit: Numeric Scope: User Category: Objective Requirement: Non-functional

Measurement	Understandability
Level	Usability
Definition	The ease with which users can understand the capabilities and operation of the services
Way to perform calculation	Not defined by the authors
Additional information	Type: Quantitative Unit: Numeric Scope: Developer Category: Objective Requirement: Non-functional

Measurement	User error protection
Level	Usability
Definition	The ability that a cloud service protects users against making errors.

Way to perform calculation	Not defined by the authors
Additional information	Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non-functional

Measurement	Version Control
Level	Usability
Definition	Controlling version concurrency and conflict
Way to perform calculation	Not defined by the authors
Additional information	Type: Quantitative Unit: Numeric Scope: Provider Category: Objective Requirement: Non-functional

TQACA - User Experience

Measurement	User Experience
Level	Root
Definition	A user, going to take services from Cloud, must consider the views of existing users for the services. Existing users of a service can describe accuracy, stability and transparency of the service in a better way.
Way to perform calculation	For the proposed framework, it is assumed that existing users rate Cloud services on level 0 to 10. Higher value of level indicates the better experience of users with that service.
Additional information	Type: Quantitative Unit: Scala Scope: User Category: Objective Requirement: Non-functional

Measurement	Cloud Customer Confidence Level
Level	User Experience
Definition	The Confidence and Fulfillment Matrix based on satisfaction level of Cloud Service as shown in Table 3 and calculated as follows: $\text{Cloud Customer Confidence Level} = \frac{\text{Number of workloads completed successfully within their budget and deadline}}{\text{Total number of workloads submitted}}$

Way to perform calculation	<p>The Confidence and Fulfillment Matrix based on satisfaction level of Cloud Service as shown in Table 3 and calculated as follows:</p> $\text{Cloud Customer Confidence Level} = \frac{\text{Number of workloads completed successfully within their budget and deadline}}{\text{Total number of workloads submitted}}$
Additional information	<p>Type: Quantitative Unit: Scala Scope: User Category: Objective Requirement: Non-functional</p>