

Mapeo Sistemático para la caracterización de los requerimientos de calidad no funcionales y de las arquitecturas para computación cuántica

Oscar Santiago López Erazo^{1,4} [0000-0002-8588-8365], Juliana Delle Ville² [0009-0007-7888-7544],
Giuliana Maltempo² [0009-0005-7441-5828], Adriana Gómez³ [0009-0008-5686-6408], Luis Freddy
Muñoz⁴ [0000-0002-8172-0530], Julio Ariel Hurtado¹ [0000-0002-2508-096], Leandro Antonelli^{2,5}
[0000-0003-1388-0337], Cesar Collazos¹ [0000-0002-7099-8131]

¹ IDIS, Universidad del Cauca

² Lifa, Facultad de Informática, Universidad Nacional de La Plata, Argentina

³ GIA, Universidad Tecnológica de Pereira, Colombia

⁴ Logiciel, Fundación Universitaria de Popayán

⁵ CAETI - Facultad de Tecnología Informática - Universidad Abierta Interamericana

Abstract. La mecánica cuántica es una de las ramas de la física que estudia la naturaleza a escalas pequeñas. En dichas escalas, la naturaleza de cada partícula tiene un comportamiento diferente y anti intuitivo, como el entrelazamiento cuántico o la superposición de estados. La computación cuántica aprovecha la mecánica cuántica y sus propiedades para realizar diferentes cómputos con velocidades superiores a las de la computación clásica; sin embargo, aún se encuentra en una etapa temprana. Este tipo de sistemas requieren de hardware y software específicos, y en el mercado existen diferentes soluciones. Para poder obtener software de gran calidad existen diferentes estándares, entre ellos el estándar ISO 25010 que puede ser aplicado al software cuántico. En este trabajo se realiza un mapeo sistemático del estado del arte para poder responder cuáles son los requerimientos no funcionales impactados por esta nueva tecnología y la tendencia que estos mismos sistemas siguen en cuanto a arquitectura. Entre los resultados obtenidos, se evidencia que los sistemas cuánticos tienden a emplear una arquitectura por capas. Además, se observa que los requerimientos más importantes hasta el momento son confiabilidad, rendimiento, escalabilidad y mantenibilidad.

Keywords: Computación cuántica, Requerimientos no funcionales, Arquitectura Software Cuántico, Mapeo Sistemático

1 Introducción

La mecánica cuántica es la rama de la física que estudia la naturaleza a escalas espaciales pequeñas, donde cada partícula tiene un comportamiento diferente y antiintuitivo frente a lo que se conoce a escala normal [1]. Una particularidad es el entrelazamiento cuántico, por el que dos partículas se entrelazan y el estado cuántico de cada una no puede describirse independientemente. Otra particularidad es que una partícula puede encontrarse en varios estados al mismo tiempo [1]. La computación cuántica utiliza hardware especializado que aprovecha estas propiedades para resolver problemas complejos que los sistemas clásicos no pueden resolver. Los sistemas cuánticos utilizan los bits cuánticos o cúbits (*qubits*) como unidad mínima de

información, los cuales pueden estar en 0, 1 o ambos estados al mismo tiempo. Los cúbits se conectan como un solo sistema mediante el enlazamiento cuántico.

El software clásico está determinado por un conjunto de requerimientos funcionales y no funcionales [2]; estos últimos definen la calidad del producto final. Los requerimientos no funcionales (NFR) son importantes al diseñar un sistema, conforme la norma ISO/IEC 25010 para la evaluación de los requerimientos de calidad del software clásico. Los NFR no solo ayudan a diseñar software de calidad, sino que también inciden en la arquitectura. La arquitectura es un aspecto relevante de cualquier sistema, ya que incide en la abstracción de componentes de hardware heterogéneos, garantiza operaciones fluidas mediante protocolos de red y contribuye a la calidad en sistema de software tradicionales o basados en paradigmas como la Internet de las cosas (IoT) y la computación cuántica [2].

Los arquitectos deben enfocarse en los NFR relevantes a su sistema. En la comunidad de la computación cuántica existe consenso en que las aplicaciones para este tipo de computación serán híbridas, es decir que algunas funciones se ejecutarán eficientemente en sistemas clásicos y otras en computadoras cuánticas [3]. Habrá, pues, una relación estrecha entre el software clásico y el cuántico, y se deberá hacer hincapié en el diseño de arquitecturas de software cuántico teniendo en cuenta su estructura de alto nivel, que involucra componentes, relaciones, interacciones, flujos de datos y requerimientos de calidad.

Los mismos NFR con que se define el software clásico también pueden encontrarse en el software cuántico. Es decir que la norma ISO/IEC 25010, utilizada normalmente para software clásico, también es aplicable al software cuántico y, del mismo modo, los NFR afectan a la arquitectura de un sistema cuántico. Como es sabido, la computación cuántica posee la capacidad de resolver problemas de alta complejidad con mayor eficiencia que la clásica. Por ejemplo, ayuda en la mejora de problemas como la optimización o la criptografía. Estas características impactan significativamente las arquitecturas de software y NFR, tales como la escalabilidad, integración, seguridad, fiabilidad y mantenibilidad, y exigen la creación de nuevas arquitecturas de software que integren elementos cuánticos y/o clásicos, es decir, que sean arquitecturas híbridas con el fin de aportar en la mejora la calidad requerida y en arquitecturas de nueva generación [4]. En este trabajo, se realiza un mapeo sistemático para identificar cuáles son los NFR más relevantes en el software cuántico y las tendencias en arquitecturas aplicadas en la computación cuántica.

Este trabajo se organiza de la siguiente manera: en la sección 2 se explica la metodología empleada para el mapeo. En la sección 3 se describe el estado del arte para el dominio de la computación cuántica. Luego, en la sección 4 se muestran los resultados obtenidos y se responden las preguntas de investigación definidas en el apartado de metodología. Finalmente, se presenta una discusión sobre los resultados (sección 5) y las conclusiones (sección 6).

2 Metodología

Para el presente trabajo se realizó un mapeo sistemático siguiendo dos enfoques: (1) la guía para revisiones sistemáticas en ingeniería de software propuesta por Kitchenham [5] y (2) la guía para realizar estudios de mapeo sistemático en ingeniería de software propuesta por Petersen *et al.* [6].

2.1 Preguntas de investigación

La investigación en computación cuántica ofrece en la actualidad incontables áreas pendientes de exploración. En este trabajo se encaran dos aspectos específicos, los requerimientos no funcionales y la arquitectura, con base en las siguientes preguntas, respectivamente:

- ¿Cuáles son los requerimientos de calidad no funcionales utilizados en el contexto de la computación cuántica?
- ¿Cuáles son las arquitecturas utilizadas en el contexto de computación cuántica?

2.2 Cadena de búsqueda

Para crear la cadena de búsqueda, se identificaron los términos principales y posibles sinónimos a fin de obtener el mayor número de estudios primarios relevantes. Se identificaron las siguientes cadenas de búsqueda principales: “Quantum Computing”, “Non Functional Requirements”, “Quantum Software Engineering”, “Quality Characteristics”, “Quantum Software Architecture”. También se identificaron los términos alternativos “NFRS” y “Architecture”. En la Tabla 1 se resumen las cadenas de búsqueda identificadas.

Tabla 1. Cadenas de búsqueda.

Términos principales	Términos alternativos
Quantum Computing	(“Non Functional Requirements” OR “NFRS” OR “Quality Characteristics”) AND (“Architecture” OR “Software Architecture”)
Non Functional Requirements	AND (“Quantum Software Engineering”)
Quantum Software Engineering	AND (“Quantum Computing”)
Quality Characteristics	
Quantum Software Architecture	

2.3 Estrategia de búsqueda

Para la búsqueda se consultaron los siguientes repositorios digitales: Scopus, Science Direct, IEEE Xplore, Springer y Google Académico. Las cadenas de búsqueda fueron aplicadas a los campos Título, Resumen y Palabras clave. Los documentos considerados en este estudio incluyen artículos de investigación, actas de conferencias, libros, informes técnicos y trabajos de grado en inglés y español publicadas entre 2019 y 2024. En la Tabla 2 se resumen los elementos de la estrategia de búsqueda.

2.4 Criterios de inclusión y exclusión

La búsqueda solo considera los estudios académicos y profesionales realizados en conferencias, revistas, reportes técnicos o trabajos de grado. Este trabajo excluye blogs personales, páginas web u obras escritas en español, a excepción de algunos trabajos en español que se incluyeron debido a su contribución a la investigación, como se

muestra en la Tabla 3. No se consideran trabajos que no se centren en la ingeniería de software, requerimientos no funcionales y arquitecturas para computación cuántica. Se excluyeron resúmenes de talleres y trabajos duplicados de la misma búsqueda en diferentes repositorios.

Tabla 2. Estrategias de búsqueda.

Ítem	Descripción
Repositorios consultados:	Scopus, Science Direct, IEEE Xplore, Springer y Google Académico
Tipo de publicación:	Artículos de investigación, Artículos de conferencias, Libros, Informes técnicos y Trabajos de grado
Campos de búsqueda:	Título, Abstract y Keywords
Idioma:	Estudios escritos en inglés o español
Periodo de publicación:	Desde 2019 hasta 2024

Tabla 3. Resumen de criterios de inclusión y exclusión.

Criterios	Reglas
Criterios de inclusión	Términos incluidos en la cadena de búsqueda. Estudios reportados en artículos, conferencias, libros, informes técnicos o trabajos de grado. Estudios escritos en inglés y español Publicaciones desde el 2019 hasta 2024. Estudios que establezcan la ingeniería de software, requerimientos no funcionales y arquitecturas para computación cuántica.
Criterios de exclusión para título y abstract	Estudios que no aborden la ingeniería de software, requerimientos no funcionales y arquitecturas para computación cuántica. Estudios que solo presentan resúmenes o contenido en diapositivas. El contenido de páginas web, blogs personales o folletos.
Criterios de exclusión para el texto completo	Estudios que presentan resúmenes de un workshop

Según Petersen *et al.* [6], la evaluación de la calidad es más esencial en las revisiones sistemáticas para determinar el rigor y la relevancia de los estudios primarios. En los mapeos sistemáticos no es necesario, aunque sí recomendable, realizar una evaluación de calidad. El uso de la clasificación de los tipos de investigación según la categoría de propuestas de solución mencionados en Wieringa *et al.* [7], contendría estudios sin evidencia empírica o científica. Aunque estos últimos no suelen incluirse en las revisiones sistemáticas, sí son importantes en los mapeos sistemáticos, pues ayudan a detectar tendencias en los temas trabajados. Por este motivo, en el presente estudio se incluyeron hallazgos con y sin evidencia empírica. Para refinar los estudios seleccionados, se utiliza la lista de chequeo propuesta por Zarour *et al.* [8], reproducida en la Tabla 4. Para cada elemento hay tres posibles respuestas: Sí = 1 punto, No = 0 puntos y Parcialmente = 0.5 puntos. Utilizando el primer cuartil ($6/3 = 2$) como punto

de corte, si una publicación obtiene un resultado menor que 2 se la excluye de la lista de estudios finales para evitar trabajos de baja calidad.

Tabla 4. Lista de chequeo de criterios de calidad.

#	Pregunta
QA1	¿El objetivo del estudio está lo suficientemente bien explicado?
QA2	¿Se explica claramente la idea, enfoque y limitaciones presentadas?
QA3	¿Se tienen en cuenta las amenazas en contra de la validez?
QA4	¿Existe una descripción adecuada del contexto en el que el estudio se llevó a cabo?
QA5	¿Están los resultados de la investigación claramente establecidos?
QA6	¿Se diseñó el estudio para lograr estos objetivos?

3 Trabajos relacionados

Al aplicar las cadenas de búsqueda mencionadas en los repositorios seleccionados, se obtuvo un total de 1144 resultados. Tras filtrar estos resultados mediante el título, abstract y conclusiones, se seleccionaron 100 resultados. Posteriormente, se aplican los criterios de inclusión, exclusión y calidad definidos en el punto anterior, se seleccionaron 15 resultados (Tabla 5). En las secciones 3.1 y 3.2 se describen los artículos seleccionados en el filtro anterior.

Tabla 5. Resumen de resultados tras aplicar la estrategia de búsqueda.

Cadena de búsqueda	Repositorio	Resultados encontrados	Título, abstract y conclusiones	Seleccionados con criterios de calidad
1	Scopus	0		
	Science Direct	13		
	Ieee Xplore	5		
	Springer	1085		
	Google Academic	41		
	Total		1144	100

3.1 Requerimientos no funcionales en computación cuántica

Saraiva *et al.* [9] proponen una discusión sobre los requerimientos no funcionales para los programas cuánticos y analizan la posibilidad de adaptar los NFR a los sistemas embebidos, teniendo en cuenta que el software se diseña específicamente para el hardware que lo ejecutará. Destacan algunos desafíos, como el acoplamiento del software al hardware, el error de lectura y la compilación dependiente al hardware.

Sodhi *et al.* [10] destacan, desde la perspectiva de la ingeniería de software, que los programas cuánticos afectan a varios NFR como la portabilidad, escalabilidad, testeabilidad y mantenibilidad. Agregan que, dada su naturaleza, los programas cuánticos se limitan actualmente a áreas donde prima el requerimiento de rendimiento.

Verduro *et al.* [11] resumen algunos problemas de NFR con respecto a los programas cuánticos y destacan la necesidad de usar modelos de calidad, métricas y herramientas integrales. Aclaran que este entorno no solo es útil para el desarrollo de software cuántico, sino que también para los objetivos de modernización y reingeniería.

Yue *et al.* [3] explican cómo podrían diferenciarse las ingenierías de requerimientos para sistemas cuánticos y clásicos. Además, los resultados de sus encuestas revelan la importancia de los métodos formales para especificar rendimientos para software cuántico. Sin embargo, dichos métodos son todavía muy rudimentarios.

Valdez *et al.* [12] realizan una revisión sobre los trabajos y aplicaciones más relevantes de computación cuántica, utilizando el estándar de calidad ISO 25010 para clasificar cada NFR del software cuántico. Resaltan que los algoritmos cuánticos se describen en términos que facilitan probar su correctitud y su derivación de estimadores de complejidad asintótica. Mencionan los desafíos de que no se consideren conexiones individuales de cúbits, o que las soluciones deban integrarse en el ciclo de vida de la ingeniería de software cuántica.

Sepúlveda *et al.* [13] presentan un protocolo de revisión sistemática para ingeniería de requerimientos basado en las guías de Kitchenham [5]. Con este protocolo buscan sintetizar las metodologías actuales para superar desafíos y proponer nuevas perspectivas a la evolución de este tipo de metodologías.

Sodhi B. [10] presenta una revisión del estado de arte para los programas de computación cuántica, en la que identifica los relevantes desde una perspectiva de la arquitectura de software. Entre los problemas que encuentra se menciona la falta de un sistema operativo cuántico nativo, el bajo nivel de abstracciones, la dependencia software-hardware, entre otros. Describe además cómo se ven afectados los requerimientos de disponibilidad, interoperabilidad, mantenibilidad, manejabilidad, performance, confiabilidad, escalabilidad, seguridad, testabilidad y usabilidad.

3.2 Arquitecturas adoptadas en computación cuántica

Entre los desafíos que identifican Arif *et al.* [14] se encuentran la transmisión segura de datos, la arquitectura centrada en procesos, las herramientas arquitectónicas y el soporte tecnológico, y el conocimiento y experiencia arquitectónica.

Ricardo Perez-Castillo *et al.* [15] presentan dos grandes desafíos para el diseño de software cuántico: (1) el modelado a alto nivel y (2) el desarrollo en sistemas de información híbridos. La solución que brindan es un perfil de UML cuántico para el análisis y diseño del sistema híbrido; no obstante las notaciones UML son insuficientes para representar las propiedades fundamentales de la computación cuántica.

Awan *et al.* [16] identifican, examinan y priorizan cambios críticos en la industria del software. Además, implementan un proceso de jerarquía analítica difusa para evaluar y graduar los desafíos o barreras encontrados, como la necesidad de escalar el hardware cuántico para que pueda competir con el clásico, la detección de errores, problemas de incoherencias, entre otros.

Dwivedi *et al.* [17] exploran los lenguajes de programación cuánticos de ingeniería de software cuántica y el ciclo de vida del desarrollo cuántico, destacando el reuso de software cuántico. El principal desafío es la falta de abstracciones de alto nivel.

Oliver *et al.* [18] identifican los desafíos que se deben enfrentar al diseñar nuevas arquitecturas de software en el contexto de la computación cuántica, a saber: que los proveedores de hardware cuántico son escasos y aplican diferentes soluciones y técnicas, y que la computación cuántica actual se centra en aplicaciones específicas y no de uso general. Consideran cómo refactorizar lo disponible para implementar una nueva arquitectura de software con roles y responsabilidades bien definidos, pero los fundamentos de la ingeniería de software son insuficientes para comenzar.

Murali *et al.* [19] analizan la arquitectura de las computadoras cuánticas en base a pruebas de benchmarks, full-stack y análisis del hardware. Implementaron la herramienta TriQ para compilar y testear diferentes lenguajes en diferentes entornos. Los resultados muestran que aprovechando los detalles del hardware en el compilador se puede dar un impulso significativo en la ejecución exitosa de los programas.

Ahmad *et al.* [20] exploran la arquitectura cuántica como servicio y la implementación de microservicios cuánticos basados en la arquitectura clásica. A partir de una encuesta con practicantes de la computación cuántica, describen tres desafíos: validación interna, generalización del diseño en otros contextos y grado de credibilidad.

Gill *et al.* [21] elaboran una taxonomía para mostrar los vacíos del estado de arte de la computación cuántica. Hallaron que existen similitudes entre varios desafíos de la computación cuántica con respecto a Internet de las Cosas (IoT).

4 Resultados

4.1 ¿Cuáles requerimientos de calidad no funcionales son utilizados en el contexto de computación cuántica? (Q1)

Como se observa en la Tabla 6, el NFR al que se da mayor importancia en el contexto de la computación cuántica es la confiabilidad, con 5 apariciones en la bibliografía consultada. También son frecuentes los requerimientos de rendimiento y escalabilidad, con 4 apariciones cada uno, y el de mantenibilidad, con tres. Otros requerimientos tratados en la bibliografía son la disponibilidad, manejabilidad, seguridad, testeabilidad, usabilidad, portabilidad, reusabilidad, eficiencia, tolerancia a fallas y optimización.

Tabla 6. Frecuencia de requerimientos no funcionales en las publicaciones analizadas

NFR	Confiabilidad	Rendimiento	Escalabilidad	Mantenibilidad	Disponibilidad	Interoperabilidad	Manejabilidad	Seguridad	Testeabilidad	Usabilidad	Portabilidad	Reusabilidad	Eficiencia	Tolerancia a fallas	Optimización
Frec.	5	4	4	3	2	2	2	2	2	2	1	1	1	1	1
Ref.	[3] [9] [10] [11] [12]	[3] [9] [10] [11]	[3] [9] [10] [11]	[3] [10] [11]	[10] [11]	[10] [11]	[10] [11]	[10] [11]	[10] [11]	[10] [11]	[3]	[3]	[9]	[9]	[12]

4.2 ¿Cuáles son las arquitecturas que se han utilizado en el contexto de computación cuántica? (Q2)

A partir del análisis de los artículos recabados, se observó que el número de arquitecturas discutidas en el contexto de la computación cuántica es limitado, con sólo tres variables. De estas, destaca la arquitectura en capas, como se ve reflejado en la Tabla 7. Seguido a esta se encuentran las arquitecturas que siguen los patrones de “tuberías” (*pipes*) y “filtro” (*filter*).

Tabla 7. Frecuencia de distintas arquitecturas en las publicaciones analizadas.

Arquitectura	Frecuencia	Referencia
En capas	3	[4] , [14], [15]
Patrón de “tuberías”	1	[14]
Patrón de “filtro”	1	[14]

5 Discusión

La computación cuántica tiene la capacidad de resolver problemas de alta complejidad con mayor eficiencia que la clásica. Esto impacta significativamente en NFR como la confiabilidad, escalabilidad, rendimiento, mantenibilidad, entre otros.

La confiabilidad es uno de los requerimientos de calidad no funcionales de mayor importancia en cualquier software. En los sistemas cuánticos, cuya naturaleza los hace sumamente sensibles a las perturbaciones externas, alcanzar un nivel óptimo de confiabilidad trae aparejados numerosos desafíos. En este sentido, es necesario encontrar soluciones para corregir errores cuánticos y minimizar la decoherencia mediante el diseño de algoritmos cuánticos robustos y el control ambiental.

Por otro lado, la importancia del NFR de rendimiento radica en que la computación cuántica tiene el potencial para acelerar exponencialmente los programas tradicionales. En la bibliografía consultada se indica que algunos problemas de complejidad NP podrían alcanzar P [22]. También son relevantes la eficiencia y la optimización de recursos para mejorar el rendimiento del sistema en términos de bits, compuertas y memoria cuánticos. Asimismo, el rendimiento se relaciona con la eficiencia energética, debido a la excesiva cantidad de energía que consumen estos sistemas. En este aspecto, existen similitudes con la gestión de recursos energéticos en IoT [21].

La escalabilidad es otro NFR que presenta desafíos debido a la diversidad de las soluciones de hardware, cada una con sus programas específicos. La marcada dependencia software-hardware impide escalar esta tecnología, de modo similar a lo que ocurre en el dominio de IoT [21]. Esto, sumado a la constante evolución del software y hardware, también incrementa la dificultad y el costo de mantenimiento de los equipos y los programas. Otro problema de escalabilidad se relaciona con la dificultad para introducir más cúbits en el sistema, lo cual es necesario para realizar operaciones más complejas.

Los NFR, que definen el comportamiento del sistema en diferentes contextos, influyen a su vez sobre decisiones de diseño arquitectónico, como estrategias, tecnologías o patrones. Las decisiones arquitectónicas también buscan reducir los errores producidos por la sensibilidad inherente a los sistemas cuánticos. Los resultados de nuestro relevamiento muestran que la arquitectura de capas aparece con mayor

frecuencia, por una amplia evidencia de quantum como servicio y al amplio abanico de soluciones con sus respectivos compiladores con hardware específico. Es pertinente destacar que el dominio de la computación cuántica ofrece posibilidades para la adopción de arquitecturas de nueva generación o la creación de nuevas arquitecturas híbridas, que integren elementos cuánticos y clásicos, con el fin de mejorar la calidad requerida [3].

6 Conclusiones

En este paper se realizó un mapeo sistemático del estado de arte de tanto como afecta los requerimientos de calidad de software como la tendencia a la arquitectura en capas de la computación cuántica. En este mapeo se evidenció que la computación cuántica tiende a utilizar la arquitectura en capas, así como también la tendencia a la utilización de la computación cuántica como un servicio. Por lo que, a nivel de organización, usar arquitectura en capas permite estructurar y organizar diferentes componentes para elaborar un sistema cuántico funcional y eficiente. Asimismo, esta arquitectura permite abordar la complejidad de estos sistemas cuánticos gracias a su organización facilitando la investigación, desarrollo e implementación de tecnologías cuánticas. También se analizó el impacto que tienen los sistemas de cómputo cuántico con respecto a los requerimientos de calidad de software. Entre los resultados de este mapeo, se observó que los requerimientos más importantes hasta ahora son la confiabilidad, el rendimiento, la escalabilidad y el mantenimiento. Con respecto a la confiabilidad, debido a la naturaleza de la mecánica cuántica, se está trabajando en solucionar el desafío de la coherencia, minimizando lo más posible el margen de error. Por otra parte, esta tecnología en términos de rendimiento brinda una velocidad que supera a la computación clásica, sin embargo, aún siguen existiendo desafíos como por ejemplo mejorar la eficiencia energética para compensar el consumo excesivo. Además, la naturaleza heterogénea del hardware cuántico hace que la escalabilidad de este tipo de sistemas cuánticos sea más difícil de garantizar. Y esto a su vez afecta a la mantenibilidad del código ya que cada hardware tiene asociado su propio software específico. Como trabajo futuro, se plantea seguir investigando sobre métodos formales que permitan el diseño de arquitecturas para computación cuántica teniendo en cuenta el impacto de los NFR en las mismas.

Referencias

1. P. A. M. Dirac, *Principios de mecánica cuántica*. 1967.
2. O. S. Lopez and J. A. Hurtado, "IoTAP: Prácticas de diseño de la arquitectura y sus requisitos de calidad en el contexto de desarrollo de software basado en IoT," 2023. [Online]. Available: <http://repositorio.unicauca.edu.co:8080/handle/123456789/8566>
3. Tao Yue, Shaukat Ali, and Paolo Arcaini, "Towards Quantum Software Requirements Engineering," *Software Engineering*, 2023.
4. T. Yue, W. Maurer, S. Ali, and D. Taibi, "Challenges and Opportunities in Quantum Software Architecture," *Software Architecture*, pp. 1–23, 2023, doi: 10.1007/978-3-031-36847-9_1.

5. S. Kitchenham, B & Charters, "Guidelines for performing systematic literature reviews in software engineering," *Technical report, Ver. 2.3 EBSE Technical Report. EBSE*, vol. 1, no. January 2007, pp. 1–54, 2007, [Online]. Available: <https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.117.471&rep=rep1&type=pdf>
6. K. Petersen, R. Feldt, S. Mujtaba, and M. Mattsson, "Systematic Mapping Studies in Software Engineering," 2008, doi: 10.1142/S0218194007003112.
7. R. Wieringa, N. Maiden, N. Mead, and C. Rolland, "Requirements engineering paper classification and evaluation criteria: a proposal and a discussion," *Requir Eng*, vol. 11, no. 1, pp. 102–107, Mar. 2006, doi: 10.1007/s00766-005-0021-6.
8. M. Zarour, A. Abran, J.-M. Desharnais, and A. Alarifi, "An investigation into the best practices for the successful design and implementation of lightweight software process assessment methods: A systematic literature review," *Journal of Systems and Software*, vol. 101, pp. 180–192, Mar. 2015, doi: 10.1016/j.jss.2014.11.041.
9. Lorenzo Saraiva, Edward Hermann Haeusler, Vaston G. Costa, and Marcos Kalinowski, "Non-Functional Requirements for Quantum Programs," in *Q-SET@QCE*, 2021.
10. B. Sodhi and R. Kapur, "Quantum Computing Platforms: Assessing the Impact on Quality Attributes and SDLC Activities," in *2021 IEEE 18th International Conference on Software Architecture (ICSA)*, IEEE, Mar. 2021, pp. 80–91. doi: 10.1109/ICSA51549.2021.00016.
11. J. Verduro, M. Rodríguez, and M. Piattini, "Software quality issues in quantum information systems," *CEUR Workshop Proc*, vol. 3008, pp. 54–59, 2021.
12. F. Valdez and P. Melin, "A review on quantum computing and deep learning algorithms and their applications," *Soft comput*, vol. 27, no. 18, pp. 13217–13236, Sep. 2023, doi: 10.1007/s00500-022-07037-4.
13. S. Sepúlveda and A. Cravero, "Systematic Review Protocol: Requirements Engineering in Quantum Computing," Nov. 2023, doi: 10.5281/zenodo.10182527.
14. A. A. Khan *et al.*, "Software architecture for quantum computing systems — A systematic review," *Journal of Systems and Software*, vol. 201, p. 111682, Jul. 2023, doi: 10.1016/j.jss.2023.111682.
15. R. Pérez-Castillo and M. Piattini, "Design of classical-quantum systems with UML," *Computing*, vol. 104, no. 11, pp. 2375–2403, Nov. 2022, doi: 10.1007/s00607-022-01091-4.
16. U. Awan, L. Hannola, A. Tandon, R. K. Goyal, and A. Dhir, "Quantum computing challenges in the software industry. A fuzzy AHP-based approach," *Inf Softw Technol*, vol. 147, p. 106896, Jul. 2022, doi: 10.1016/j.infsof.2022.106896.
17. K. Dwivedi, M. Haghparast, and T. Mikkonen, "Quantum software engineering and quantum software development lifecycle: a survey," *Cluster Comput*, Mar. 2024, doi: 10.1007/s10586-024-04362-1.
18. J. L. Hevia Oliver, "Requirements for quantum software platforms," *CEUR Workshop Proc*, vol. 2705, pp. 20–26, 2020.
19. P. Murali, N. M. Linke, M. Martonosi, A. J. Abhari, N. H. Nguyen, and C. H. Alderete, "Full-stack, real-system quantum computer studies," in *Proceedings*

- of the 46th International Symposium on Computer Architecture*, New York, NY, USA: ACM, Jun. 2019, pp. 527–540. doi: 10.1145/3307650.3322273.
20. A. Ahmad, A. B. Altamimi, and J. Aqib, “A Reference Architecture for Quantum Computing as a Service,” no. i, 2023, [Online]. Available: <http://arxiv.org/abs/2306.04578>
 21. S. S. Gill *et al.*, “Quantum computing: A taxonomy, systematic review and future directions,” *Softw Pract Exp*, vol. 52, no. 1, pp. 66–114, Jan. 2022, doi: 10.1002/spe.3039.
 22. H. P. Paudel *et al.*, “Quantum Computing and Simulations for Energy Applications: Review and Perspective,” *ACS Engineering Au*, vol. 2, no. 3, pp. 151–196, Jun. 2022, doi: 10.1021/acengineeringau.1c00033.