



## Calidad en el desarrollo de software desde una perspectiva bibliométrica en el periodo 2018 – 2022

## Quality in software development from a bibliometric perspective in the period 2018 – 2022

Darío Javier Robayo-Jácome

*Pontificia Universidad Católica del Ecuador, Ambato, Ecuador*

[drobayo@pucesa.edu.ec](mailto:drobayo@pucesa.edu.ec)

 <https://orcid.org/0000-0002-0661-6573>

Recepción: 19/06/2023 | Aceptación: 24/08/2023 | Publicación: 10/09/2023

### Cómo citar (APA, séptima edición):

Robayo-Jácome, D. J. (2023). Calidad en el desarrollo de software desde una perspectiva bibliométrica en el periodo 2018 - 2022. *INNOVA Research Journal*, 8(3), 173-188.  
<https://doi.org/10.33890/innova.v8.n3.2023.2299>

### Resumen

La investigación muestra el análisis bibliométrico de la ingeniería de software, situación que aparece de la necesidad de entender cómo ha ido evolucionando el proceso de desarrollo y cómo las empresas lo han ido adoptando, adaptando y estandarizando para la construcción de software de calidad. El objetivo del trabajo es realizar un estudio bibliométrico de la producción científica de la ingeniería de software orientada a la calidad, cuyos resultados sean base para futuras investigaciones. La metodología utilizada fue descriptiva – retrospectiva, siendo los datos recolectados a partir de la base de datos Scopus; para su obtención y posterior análisis, se utilizaron varios parámetros que definieron los criterios de inclusión y exclusión, obteniéndose 144 artículos resultantes; además, se plantearon múltiples interrogantes que permitieron visualizar las principales tendencias a través del análisis descriptivo de los mismos. Al final del proceso, se determinó la revista IEE Access con más artículos publicados, Mkaouer Mohamed Wiem como el autor más relevante en el tema, Estados Unidos como el país con el mayor número de publicaciones, entre otros indicadores. Además, se evaluó la importancia de este tipo de investigaciones y las preferencias de los investigadores, todo ello, evidenciado por el número de publicaciones que se han desarrollado en el periodo 2018 – 2022.

**Palabras claves:** ingeniería de software; calidad del software; desarrollo de software; análisis bibliométrico.

## Abstract

The research shows the bibliometric analysis of software engineering, a situation that arises from the need to understand how the development process has been evolving and how companies have been adopting, adapting, and standardizing it for the development of quality software. The objective of this work is to carry out a bibliometric study of the scientific production of quality-oriented software engineering, whose results will be the basis for future research. The methodology used was descriptive - retrospective, being the data collected from the Scopus database; For its collection and subsequent analysis, several parameters were used to define the inclusion and exclusion criteria, obtaining 144 resulting articles; in addition, multiple questions were raised that allowed visualizing the main trends through their descriptive analysis. At the end of the process, the IEE Access journal with the most published articles was determined, Mkaouer Mohamed Wiem as the most relevant author on the subject, the United States as the country with the highest number of publications, among other indicators. In addition, the importance of this type of research and the preferences of researchers were evaluated, all evidenced by the number of publications that have been developed in the period 2018 – 2022.

**Keywords:** software engineering; software quality; software development; bibliometric analysis.

## Introducción

La Ingeniería de Software es un campo de investigación y desarrollo en constante evolución que busca garantizar que el software cumpla con altos estándares de calidad y confiabilidad (Carrizo y Alfaro, 2018). En los últimos 10 años, este tema ha ganado un interés significativo tanto en la comunidad académica como en la industria, como respuesta a la creciente demanda de aplicaciones confiables libres de defectos.

El desarrollo de software ha sido constantemente cuestionado debido a que las aplicaciones no responden a las expectativas de los usuarios, fallan con cierta frecuencia, los costos son difíciles de prever ya que superan las estimaciones, se entrega fuera del tiempo establecido, entre otras (Delgado Olivera y Diaz Alonso, 2021). En tal sentido, desde el apareamiento de la Ingeniería de Software, ésta ha ayudado a resolver los problemas mencionados, lo que conlleva a que el desarrollo de aplicaciones se ejecute de una manera eficiente, organizada, ordenada y disciplinada; con lo que se logra una mayor satisfacción en los usuarios finales.

En el presente artículo, se muestra un análisis bibliométrico realizado en la base de datos Scopus, con el objetivo de identificar las tendencias y perspectivas actuales en el uso de la Ingeniería de Software orientada al desarrollo de aplicaciones de calidad. El análisis bibliométrico es una metodología que utiliza técnicas estadísticas para cuantificar y analizar la producción científica en un área de conocimiento específica, proporciona información valiosa para la identificación de patrones, tendencias y áreas de investigación emergentes (Jaimes Hernández, 2020).

La investigación se centra en artículos científicos publicados en revistas y conferencias indexadas en Scopus durante el periodo 2018 – 2022, a través de criterios de selección predefinidos se extraen los documentos relevantes que abordan temáticas relacionadas con el tema de investigación. En este artículo, se analizaron diversas métricas bibliométricas, como la producción científica por año, las revistas más influyentes en el campo, los autores más prolíficos, las palabras

clave con mayor recurrencia y las tendencias temáticas identificadas. Además, se realizó un análisis de la colaboración científica y las redes de coautoría, para identificar los principales centros de investigación y las relaciones entre los investigadores.

Este análisis proporciona una visión global del estado actual de la investigación en Ingeniería de Software, identificando los avances más significativos y las áreas en las que se requiere una mayor atención. Los resultados permitirán a los investigadores, profesionales y académicos comprender las tendencias actuales y las perspectivas futuras en este campo, fomentando la colaboración y el desarrollo de nuevas investigaciones orientadas a mejorar la calidad del software.

### **Marco Teórico**

En palabras de Sommerville (2011) la Ingeniería de Software es una disciplina que se interesa por todos los aspectos de la producción de software. Estos aspectos incluyen la comunicación, la planeación, el modelado, la construcción y el despliegue, que se ejecutan a través de cierto número de repeticiones dentro del proyecto, en dependencia con la metodología de desarrollo a aplicarse (Pressman, 2010). Las metodologías de desarrollo son una especie de colección estructurada de actividades que ofrecen a los desarrolladores de software una guía para la planificación, gestión, control y evaluación. La elección de la metodología a emplear es clave durante el desarrollo de aplicaciones por sus implicaciones en lo referente a efectividad, eficiencia y desempeño del producto, costo y el tiempo de desarrollo, métodos de control de calidad y de pruebas, los cuales, deben ajustarse a las particularidades de cada metodología (Velásquez, et al., 2019). Las metodologías de desarrollo de software se clasifican en tradicionales y modernas, estas denominadas ágiles por su característica de tener tiempos cortos de entrega de las aplicaciones y sus componentes.

Para Piattini Velthuis (2016) la Ingeniería de Software es una disciplina joven con relación a otras de su misma naturaleza, ya que, ha recorrido algunas veces caminos poco claros, siguiendo prácticas empíricas sin saber muy bien a dónde conducían. Sin embargo, como menciona García Peñalvo (2018), esta ha evolucionado rápidamente, lo que supone una auténtica revolución en el mundo del software, ya que, se ha pasado de una codificación artesanal a un desarrollo sistemático de programas. Este incluye también la realización de pruebas, una actividad iterativa, cara, lenta y compleja en el desarrollo de software; no obstante, es necesaria ya que una ausencia de comprobación o una comprobación ineficaz y poco eficiente puede dar lugar a una producción de mala calidad (Dalal, Solanki, Sudhir, y Diksha, 2019).

Como fruto de esta evolución, las técnicas de desarrollo actualizadas permiten la gestión de sistemas grandes y complejos, enfrentar el cambio constante de los requerimientos, poner énfasis en las propiedades no funcionales, ver y ofertar el software como servicio, y unirse al desarrollo con la aplicación de metodologías ágiles.

Sin embargo, los problemas en ingeniería de software aún están estrechamente ligados a la falta de un cuerpo de conocimiento que más que amplio debe ser concreto en casos particulares de desarrollo. Pensar que solo con el uso de procesos se obtiene un producto de calidad, es una idea errada, ya que a pesar de estar basado en un conjunto de principios que formalizan el desarrollo,

no deja de lado la responsabilidad en las técnicas y habilidades personales de los desarrolladores. Otros de los problemas aún presentes que afectan significativamente a los proyectos de desarrollo de software es la falta de predictibilidad para su ejecución dentro del tiempo y presupuesto proyectado; esto como consecuencia de los errores en las estimaciones de esfuerzo o de la utilización de metodologías de desarrollo equivocadas o en una gestión mal aplicada (Pantaleo y Rinaudo, 2016).

Algo que preocupa a los ingenieros es el transcurso del tiempo y la inminente necesidad de cambios, por ello, en mayor medida se debe considerar la escalabilidad y la eficiencia, tanto en lo que se refiere al software que se crea como al equipo encargado de crearlo. Para Winters, Manshreck, y Wright, (2022), se debe considerar la necesidad de delimitar entre tareas de programación (desarrollo) y tareas de ingeniería de software (desarrollo, integración y mantenimiento), y si a ello, se le añade el factor tiempo, da lugar a una nueva dimensión tanto a la programación como al mismo hecho de todo lo que involucra el desarrollo de un producto de software. Tampoco se deben dejar de lado el factor costo, ya que el alcance de la calidad del producto de software que los modelos de calidad son capaces de evaluar tiene un impacto directo en los costes de desarrollo y mantenimiento (Galli, Chiclana, y Siewe, 2021).

Por otra parte, el software al igual que otros productos presenta diferentes enfoques para la valoración de su calidad, pero no es valorable desde aspectos como resistencia de sus materiales, dimensiones, tiempo de vida útil, entre otros (Gómez Palomo y Moraleda Gil, 2020); lo que dificulta hacer una evaluación mucho más objetiva de cuanto calidad presenta una aplicación. De acuerdo con Pantaleo y Rinaudo (2016), el desafío actual con respecto a garantizar la calidad es organizarse para que sus proyectos sean ordenados y predecibles, sin olvidarse que deben también tener la capacidad para adaptarse a los cambios; por ello, es importante lograr una madurez en los procesos de manera que estén estandarizados y permitan la consecución de los proyectos produciendo software de calidad. Es así como la calidad en el desarrollo de software se refiere al nivel de cumplimiento de las principales características durante su ciclo de vida, para brindar el mejor servicio a los usuarios (Sánchez Jaya, Robayo Jácome, y López Sevilla, 2018).

A lo anterior, se agrega que la entrega de software funcional en corto tiempo es una forma de agregar valor a los clientes, lo que está fuertemente respaldado por los principios de la gestión del valor que involucran el tiempo y el riesgo (Ruiz de Mendarozqueta, Bustos, y Colla, 2022). Para ello, el uso de metodologías ágiles acorta los tiempos de entrega lo que no necesariamente garantiza calidad, por ejemplo, en términos de Alami y Krancher (2022) el valor añadido de Scrum para la calidad del software no sustituye a las buenas prácticas de ingeniería y garantía de calidad, por el contrario, los complementa para elevar la capacidad de los equipos de desarrollo para lograr calidad en el software.

De acuerdo con Veloz Segura (2022), la presencia y uso de estándares y normas en el desarrollo de software, ha propuesto estrategias y guías que han dado lugar a la estandarización de la calidad, como apoyo a la construcción y al uso de un producto software; esto permite evaluar si el nivel de calidad durante su ciclo de vida es aceptable y de esta manera prever la calidad de la aplicación final. Por otra parte, y para mejorar el proceso de desarrollo de software en términos de calidad, hay que considerar el aspecto emocional de las personas implicadas, ya que por ejemplo

una pronta identificación de posibles emociones negativas podría evitar un impacto negativo en la calidad del producto final (Khan y Saleh, 2021).

Finalmente, es importante mencionar la definición de calidad de software descrita en el estándar ISO/IEC TS 25011:2017 (ISO/IEC, 2017), citado en Peláez Valencia, Hurtado Gil, Toro Lazo, y Arias Vargas (2018), la cual, dice que la calidad es “el grado en el cual un producto de software satisface las necesidades establecidas e implícitas, cuando es utilizado en condiciones específicas” (p. 20), misma que, incluye un elemento importante: las necesidades del usuario.

## Metodología

El estudio se realiza mediante una investigación con enfoque cuantitativo, por los números que la bibliometría arroja al comparar la evolución y contribución de varios autores sobre la temática (Torres, 2016). Para llevar a cabo el análisis bibliométrico en el ámbito de la Ingeniería de Software orientada a la calidad de las aplicaciones, se usa un método de tipo descriptivo con un enfoque cuantitativo para realizar un análisis documental, sustentado en herramientas bibliométricas para análisis de redes y generación de gráficos.

La técnica de muestreo empleada fue el muestreo intencional ya que permite seleccionar sujetos que cumplan con las características necesarias para la investigación (Otzen y Manterola, 2017). La búsqueda se realizó en Scopus, la cual, es una de las bases de datos con mayor cobertura cuyas publicaciones son de la más alta calidad científica (Bartels, 2013), con esta búsqueda inicial al hacer uso de las palabras clave *software quality* y *software quality assurance*, se obtuvo un total de 572 artículos.

Posteriormente, se definieron los criterios de inclusión, mismos que sirven para determinar los artículos seleccionados para el análisis; así mismo, se definieron los criterios de exclusión para delimitar los artículos que no formarán parte del trabajo:

### Criterios de inclusión:

- Artículos que contengan las palabras clave:
  - software engineering,
  - software quality,
  - software quality assurance.
- Publicaciones realizadas en el periodo 2018-2022.
- Artículos científicos a texto completo.

### Criterios de exclusión:

- Artículos en idioma diferente al inglés.
- Resúmenes, ponencias, actas de conferencia.

Con base a los criterios mencionados, se definió la siguiente cadena de búsqueda:

```
TITLE-ABS-KEY (("software quality" AND "software engineering") OR ("software quality assurance" AND "software engineering")) AND ( ( PUBYEAR = 2018 ) OR ( PUBYEAR = 2019 ) OR ( PUBYEAR = 2020 ) OR ( PUBYEAR = 2021 ) OR ( PUBYEAR = 2022 ) ) AND ( LIMIT-TO ( SUBJAREA , "COMP" ) ) AND ( LIMIT-TO ( DOCTYPE , "ar" ) ) AND ( LIMIT-TO ( LANGUAGE , "English" )) AND ( LIMIT-TO ( EXACTKEYWORD , "Software Engineering" ) OR LIMIT-TO ( EXACTKEYWORD , "Software Quality" ))
```

Al aplicar los criterios de inclusión y exclusión mediante la cadena de búsqueda, se obtuvo un total de 144 documentos, que cumplen con los requerimientos para el análisis bibliométrico. Estos documentos se organizaron en una hoja de cálculo de Microsoft Excel, incluyendo datos como nombres de los autores, título del documento, año de publicación, afiliación de los autores, revista de publicación, país de publicación, número de citas recibidas, entre otros.

De los resultados se extrajeron los metadatos para posteriormente ser analizados por medio del software de análisis de redes bibliométricas VOSviewer, el cual usa para la construcción de mapas la técnica denominada VOS (visualization of similarities), que considera información importante como la citación y cocitación de autores y revistas, así como la coocurrencia de palabras clave y textos (Van Eck, Waltman, Dekker, y Van Den Berg, 2010). Además, los datos de las publicaciones seleccionadas se procesaron también en la aplicación ScienceScape para un análisis adicional y la generación de gráficos descriptivos.

De la misma manera, se realizó también una revisión sistemática de la literatura hasta el año 2018, para proceder a la búsqueda y extracción de información. Gracias a esta revisión se conocen ciertos indicadores bibliométricos, como número de artículos, autores principales, revistas con más publicaciones, año de publicación, entre otros (Solano, Castellanos, López, y Hernández, 2019). Sin embargo, algunos de estos indicadores no son fáciles de interpretar, por la cantidad de publicaciones que se generan y que van en aumento año tras año (Génova, Astudillo, y Fraga, 2016).

Con base a lo anterior, se formularon las siguientes líneas investigativas:

- Número de publicaciones anuales en el periodo de revisión.
- Países con mayor número de publicaciones.
- Revistas con la mayor cantidad de artículos.
- Palabras clave más utilizadas.
- Número de artículos por autor.
- Artículos con mayor cantidad de citas.

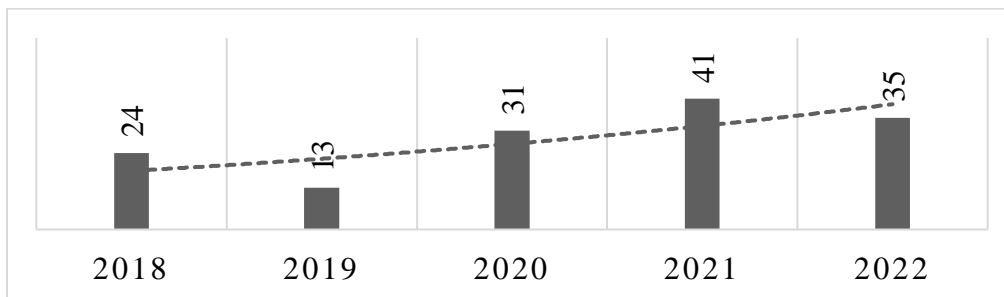
## Resultados y Discusión

Dentro del periodo analizado, si bien, se evidencia que el número de publicaciones no se ha mantenido constante respecto de cada año, denota tener un crecimiento progresivo si se considera una media en cuanto a las publicaciones anuales (Figura 1). El año 2021 presenta el mayor número de publicaciones, en contraste al año 2019 que es el menos productivo. Además, se ve la tendencia de crecimiento anual de las investigaciones de ingeniería de software orientadas a

la calidad del producto, lo que demuestra que el desarrollo de software ha experimentado un crecimiento exponencial en los últimos años y seguirá evolucionando durante el presente año y demás (Valdovinos, 2023).

### Figura 1

*Artículos publicados por año*

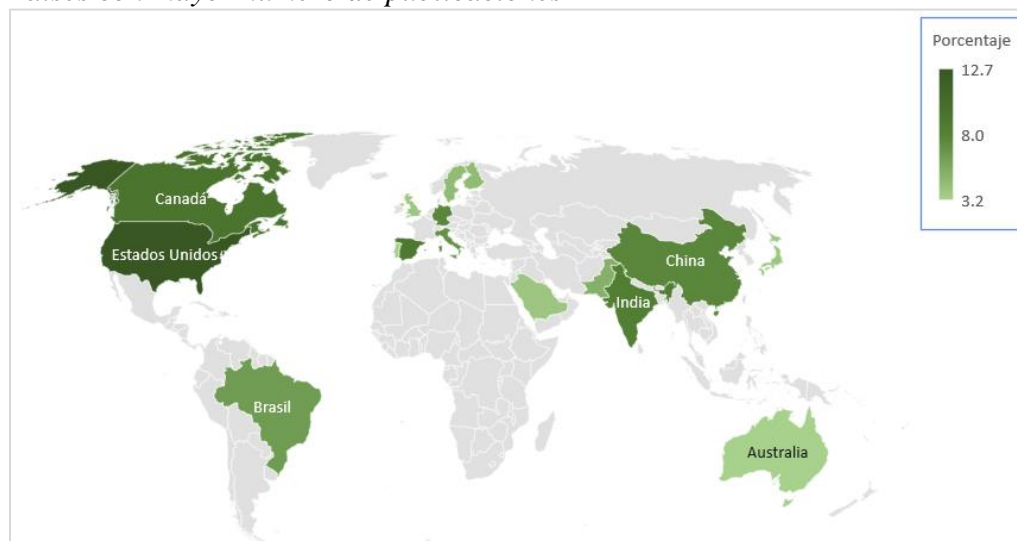


Fuente: Elaboración propia.

Desde el punto de vista de la ubicación geográfica, se aprecia que los 144 documentos revisados se reparten entre 14 países, siendo Estados Unidos quien lidera con un 12.7% de artículos publicados en el periodo 2018 – 2022 (Figura 2), seguido de Canadá con un 9.6%, en tercer lugar, se encuentran España e India con un 8%, como se aprecia en la Figura 2. En el resto de los países con más publicaciones están: Brasil, China, Alemania, Italia, entre otros, con un menor porcentaje. No puede ser una simple coincidencia que los países que lideran la producción de software a nivel mundial se encuentren dentro del ranking de países más innovadores, encontrándose en este listado a Estados Unidos, Canadá, India y China (OMPI, Organización Mundial de la Propiedad Intelectual, 2022).

### Figura 2

*Países con mayor número de publicaciones*

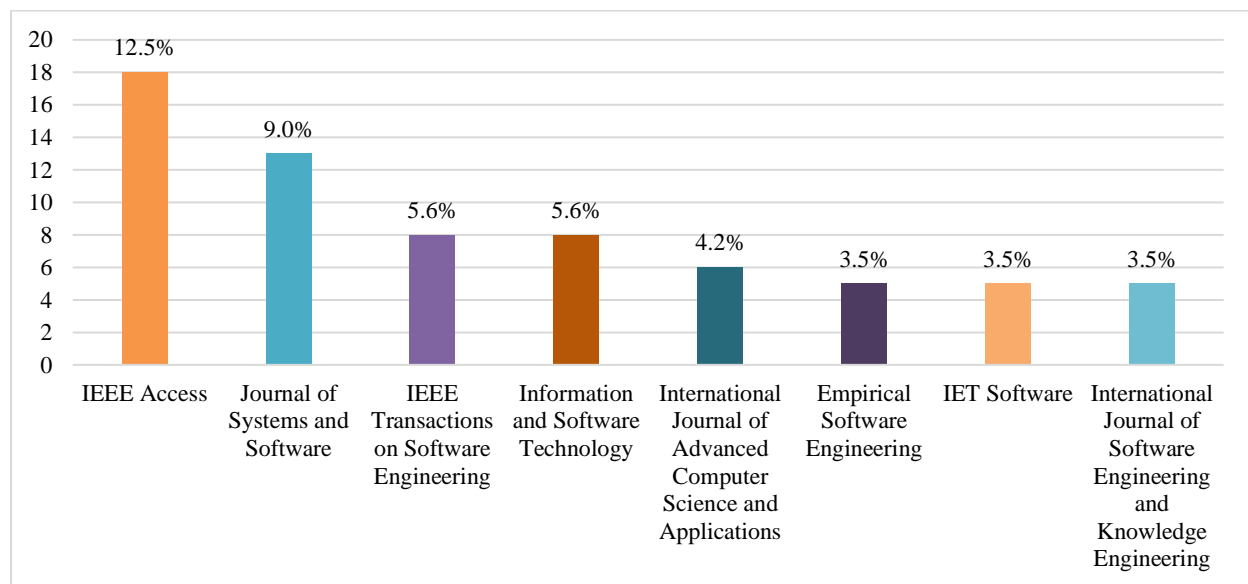


Fuente: Elaboración propia.

Las investigaciones en el área de la Ingeniería del Software y de la Calidad del Software se concentran en 69 revistas a nivel mundial. De ellas, 8 agrupan el 47.4% de todas las publicaciones, siendo estas IEEE Access con el 12.5%, Journal of Systems and Software con el 9%, IEEE Transactions on Software Engineering con el 5.6%, International Journal of Advanced Computer Science and Applications con el 4.2% y, Empirical Software Engineering, IET Software e International Journal of Software Engineering and Knowledge Engineering con el 3.5%, según se aprecia en la Figura 3. Resulta evidente que el Instituto de Ingeniería Eléctrica y Electrónica (IEEE, sigla en inglés), lidera las investigaciones en esta área con dos de sus revistas dentro de las tres primeras, lo que demuestra la calidad y el nivel científico de sus publicaciones; en el caso de IEEE Access presenta un valor Research Impact Score de 25.2 (Research.com, 2023) e IEEE Transactions on Software Engineering tiene un valor Research Impact Score de 11.8 (Research.com, 2023).

**Figura 3**

*Revistas*



*Fuente:* Elaboración propia

Desde la perspectiva de los autores y el número de publicaciones realizadas, en la Tabla 1 se observa que los autores Mkaouer Mohamed y Ouni Ali presentan la mayor cantidad de artículos relacionados con el tema, con 5 cada uno; seguido por Franch Xavier con 4 documentos y, finalmente, Graziotin Daniel, Vollmer Anna y Wagner Stefan con 3 publicaciones cada uno. Se puede apreciar que la afiliación de estos autores se da en instituciones de Norteamérica y Europa, lo que demuestra la investigación con mira a mejorar la calidad de los productos de software.



**Tabla 1**

*Número de artículos por autor*

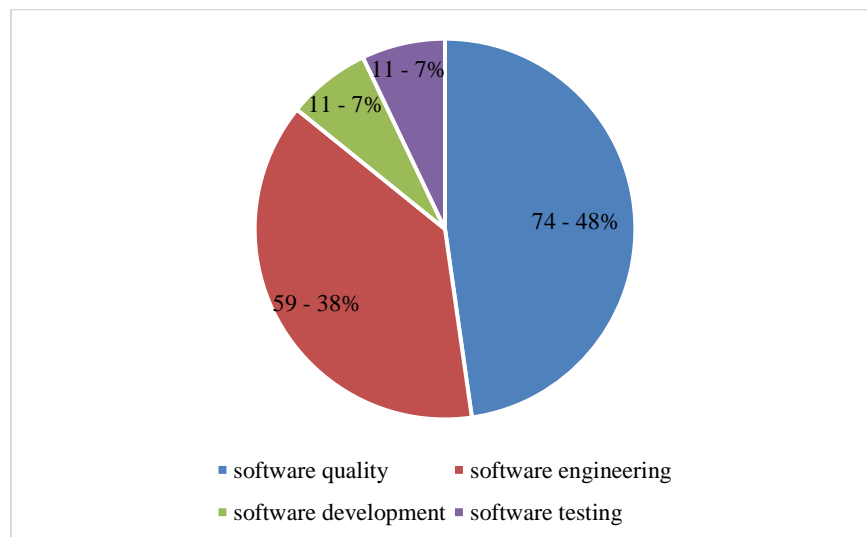
| <b>Autor</b>         | <b>Afiliación</b>   | <b>No. Artículos</b> |
|----------------------|---|----------------------|
| Mkaouer Mohamed Wiem | Rochester Institute of Technology, Estados Unidos.                    | 5                    |
| Ouni Ali             | ETS Montreal, University of Quebec, Canadá.                           | 5                    |
| Franch Xavier        | Department of Service and Information System Engineering, España.     | 4                    |
| Graziotin Daniel     | Institute of Software Technology, University of Stuttgart, Alemania.  | 3                    |
| Vollmer Anna Maria   | Fraunhofer IESE, Alemania.  | 3                    |
| Wagner Stefan        | University of Stuttgart, Institute of Software Engineering, Alemania. | 3                    |

*Fuente:* Elaboración propia.

Por otra parte, en la Figura 4 y con respecto a las palabras clave, se evidencia que la palabra que se repite con mayor frecuencia es *software quality*, misma que se encuentra en 74 documentos, seguida de *software engineering* con 59 documentos, luego se encuentran *software development* y *software testing* las cuales aparecen en 11 artículos cada una. Es evidente que la calidad en el de las aplicaciones es un factor que necesariamente debe ser incluido en los procesos de desarrollo de software sean estos empíricos o estandarizados.

**Figura 4**

*Palabras clave más utilizadas*



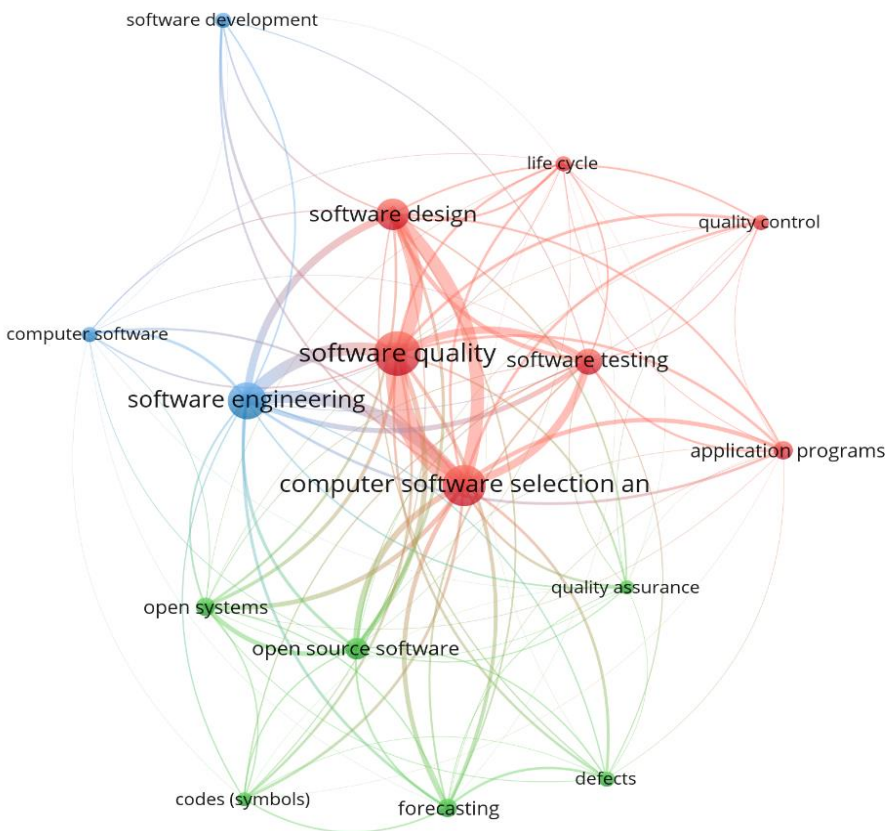
*Fuente:* Elaboración propia.

En la Figura 5, se observa el mapa de red de coocurrencia de palabras clave, la muestra de artículos analizados incluye 1305 palabras clave, de las cuales se incluyen aquellas que tienen un mínimo de 10 ocurrencias, generando una red con 16 palabras clave y 528 enlaces con una fortaleza total de 1894. En el mapa de red, los elementos se representan por su etiqueta y por un círculo, siendo su tamaño determinado por el peso del artículo, es decir, la cantidad de documentos en los que aparece la palabra clave; estos se conectan mediante líneas que representan el vínculo entre etiquetas, el cual se mide a partir de las veces que son incluidas mutuamente en documentos de la muestra. La fortaleza del vínculo está representada por el grosor de la línea, presentando líneas más gruesas si la fortaleza es mayor y líneas más delgadas si la fortaleza es menor (Nees Jan van y Waltman, 2023). Para la construcción de los clústeres y al ser solamente 16 términos clave los resultantes, se parametrizó cada clúster con un tamaño mínimo de 1 componente, obteniendo como resultado tres clústeres, en los cuales se observa un área de investigación bien definida:

- i. Calidad en el ciclo de vida del software.
- ii. Aseguramiento de la calidad en la programación de sistemas.
- iii. Uso de Ingeniería de Software en el desarrollo de aplicaciones.

## Figura 5

### *Mapa de red de coocurrencia de palabras clave*



*Fuente:* Obtenida con VosViewer.

En la Tabla 3, se detallan las ocurrencias de las palabras clave, se observa que las tres primeras palabras son: *software quality* con 114 ocurrencias, *computer software selection and evaluation* con 96 ocurrencias y *software engineering* con 77. Con ello, se evidencia que las publicaciones revisadas en el periodo de estudio están directamente relacionadas a garantizar la calidad del software en los procesos de ingeniería, lo que se aprecia también al observar las siguientes palabras clave con mayor ocurrencia como *software design* y *software testing*.

**Tabla 2**

*Ocurrencias de palabras clave*

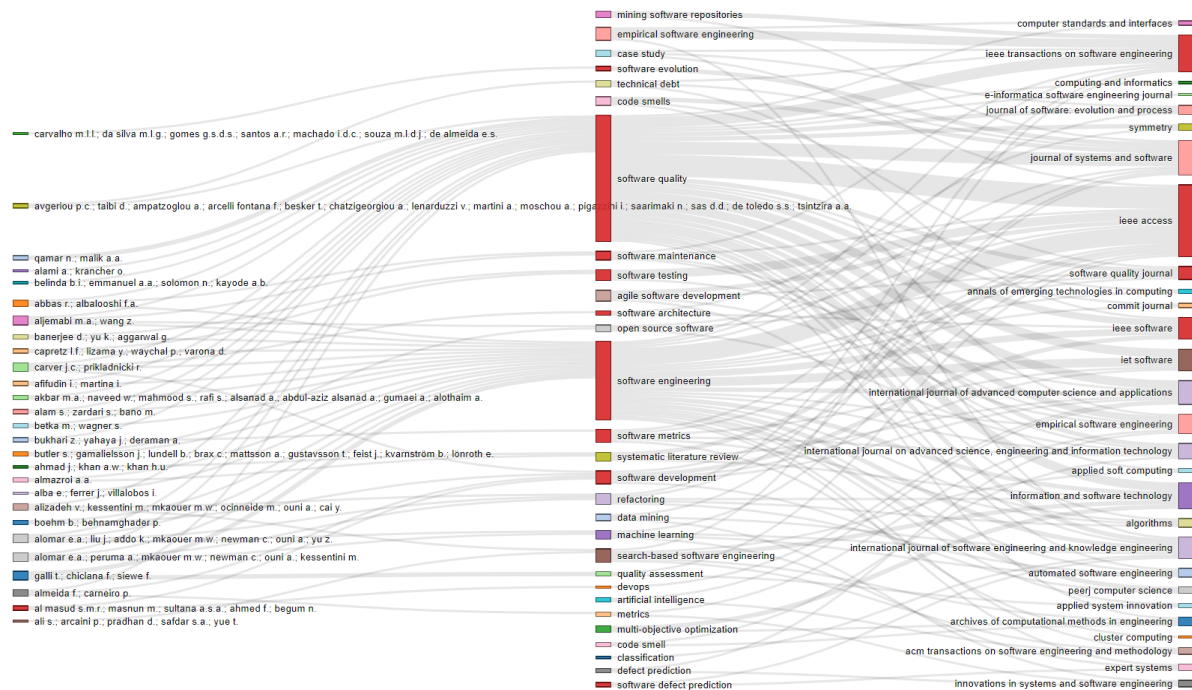
| Palabra Clave                              | Ocurrencias |
|--|-------------|
| software quality                           | 114         |
| computer software selection and evaluation | 96          |
| software engineering                       | 77          |
| software design                            | 54          |
| software testing                           | 36          |
| open source software                       | 24          |
| application programs                       | 17          |
| forecasting                                | 17          |
| open systems                               | 17          |
| life cycle                                 | 12          |
| computer software                          | 11          |
| defects                                    | 11          |
| quality control                            | 11          |
| software development                       | 11          |
| codes (symbols)                            | 10          |
| quality assurance                          | 10          |

*Fuente:* Elaboración propia.

Como complemento a lo anterior, se observa que la Figura 6 muestra las relaciones existentes entre autores, palabras clave y revistas de las publicaciones revisadas; en ella, se evidencia nuevamente el gran número de investigaciones desarrolladas respecto de la calidad del software en el proceso de ingeniería.

### Figura 6

Relación entre autores, palabras clave y revistas respecto a las publicaciones



Fuente: Obtenida con ScienceScape.

Finalmente, si se revisan las publicaciones con más relevancia considerando el número de citas, el artículo que presenta la mayor cantidad es “A survey on software smells”, con un total de 134 lo que la hace la publicación más representativa, esto se da ya que el texto trata sobre cómo los problemas de calidad en el código fuente afectan la calidad del software y dificultan su mantenimiento y evolución. Le sigue el artículo “What happens when software developers are (un) happy” con 98 citas, éste muestra los resultados basados en análisis cualitativo de más de 300 experiencias de desarrolladores respecto del impacto de la felicidad y de la infelicidad en la productividad y la calidad del desarrollo. El tercer artículo con 86 citas es “MULTI: Multi-objective effort-aware just-in-time software defect prediction” siendo ampliamente referenciado ya que se enfoca en analizar el método de predicción de defectos en el software MULTI concluyendo que funciona significativamente mejor que otros métodos de última generación para crear modelos JIT-SDP (just-in-time software defect prediction). Otro artículo destacado respecto de esta métrica, con 81 citas es “An Ensemble Oversampling Model for Class Imbalance Problem in Software Defect Prediction”, el cual hace énfasis en el uso de modelos de datos sobre muestreados para evitar el desequilibrio en la predicción de defectos del software. Los demás documentos mencionados en la tabla presentan un número considerable de citas, sin embargo, se alejan notablemente de los cuatro primeros. Es importante mencionar que de los ocho artículos considerados en la Tabla 3, cinco se han publicado en la revista “Journal of Systems and Software”.

**Tabla 3***Número de citas por artículo*

| <b>Título</b>   | <b>Autor</b>   | <b>Revista</b>                      | <b>Año</b> | <b>Citas</b> |
|---|--|-------------------------------------|------------|--------------|
| A survey on software smells   | Sharma T., Spinellis D.  | Journal of Systems and Software     | 2018       | 134          |
| What happens when software developers are (un)happy   | Graziotin D., Fagerholm F., Wang X., Abrahamsson P.                                      | Journal of Systems and Software     | 2018       | 98           |
| MULTI: Multi-objective effort-aware just-in-time software defect prediction   | Chen X., Zhao Y., Wang Q., Yuan Z.   | Information and Software Technology | 2018       | 86           |
| An Ensemble Oversampling Model for Class Imbalance Problem in Software Defect Prediction                            | Huda S., Liu K., Abdelrazek M., Ibrahim A., Alyahya S., Al-Dossari H., Ahmad S.          | IEEE Access                         | 2018       | 81           |
| A systematic literature review of blockchain and smart contract development: Techniques, tools, and open challenges | Vacca A., Di Sorbo A., Visaggio C.A., Canfora G.   | Journal of Systems and Software     | 2021       | 58           |
| Usability feature extraction using modified crow search algorithm: a novel approach                                 | Gupta D., Rodrigues J.J.P.C., Sundaram S., Khanna A., Korotaev V., de Albuquerque V.H.C. | Neural Computing and Applications   | 2020       | 54           |
| Code smells and refactoring: A tertiary systematic review of challenges and observations                            | Lacerda G., Petrillo F., Pimenta M., Gueheneuc Y.G.                                      | Journal of Systems and Software     | 2020       | 51           |
| Characterizing the contribution of quality requirements to software sustainability                                  | Condori-Fernandez N., Lago P.  | Journal of Systems and Software     | 2018       | 50           |

*Fuente:* Elaboración propia.

### Conclusiones

El uso de la bibliometría mediante la visualización de gráficos facilita reconocer la producción científica dentro de las áreas de conocimiento estudiadas, lo que ofrece una visión particular y específica de las tendencias en investigación desarrolladas por los autores.

En cuanto a la producción científica de la ingeniería de software orientada a la calidad de las aplicaciones publicada en la base de datos Scopus, se evidencia que desde el año 2018 ha crecido de manera progresiva hasta el año 2022, esto demuestra que los investigadores buscan nuevas técnicas y mecanismos de desarrollo de aplicaciones enfocados principalmente en garantizar la satisfacción del usuario final como una medida de la calidad.

En cuanto a rendimiento anual el 2021 fue el más productivo en cuanto a publicaciones; el país que más destaca fue Estados Unidos, la revista con mayor productividad de publicaciones es IEEE Access, y el autor con más artículos es Mkaouer Mohamed Wiem. En América Latina, Brasil es el único país que se destaca con mayor cantidad de estudios en ingeniería de software, lo que evidencia que los demás países no tienen publicaciones del tema en Scopus sino únicamente en bases de datos regionales.

El presente estudio se direccionó a artículos publicados únicamente en inglés ya que es el idioma que predomina en las publicaciones indizadas en Scopus, además se ignoraron documentos como libros, conferencias, tesis, cartas al editor, entre otros, para lograr uniformidad en los resultados.

Los resultados permiten apreciar una tendencia creciente de esta área de investigación en los últimos 5 años, ya que es estrictamente necesaria la producción de aplicaciones de software de calidad, misma que, debe ser considerada desde el proceso de desarrollo como tal para que el producto resultante muestre un alto nivel de calidad para el usuario. Sin embargo, al estar el trabajo limitado únicamente a publicaciones realizadas en Scopus, es recomendable realizar análisis similares en otras bases de datos de carácter científico como Web of Science, para hacer comparativas de la tendencia investigativa en el área de estudio.

Para concluir, se observa que los usuarios finales están más familiarizados con la tecnología de software, por lo que, el desarrollo de aplicaciones tendrá exigencias mayores, particularmente en la personalización de los programas, exigiendo una mayor calidad en la experiencia del usuario. En ese sentido, quienes se encuentran dentro de este proceso, deben mantenerse en constante aprendizaje no solo de la parte técnica sino también de nuevas formas de acercamiento a los usuarios, para obtener de mejor manera los requerimientos funcionales y no funcionales como base para la construcción de aplicaciones altamente satisfactorias, siendo una de las principales y mejores fuentes de aprendizaje continuo las bases de datos científicas.

### Referencias Bibliográficas

- Alami, A., & Krancher, O. (2022). How Scrum adds value to achieving software quality? *Empirical Software Engineering*, (27). <https://doi.org/10.1007/s10664-022-10208-4>
- Bartels, E. (2013). How to perform a systematic search. *Best Practice & Research: Clinical Rheumatology*, 27(2), 295-306. <https://doi.org/10.1016/j.berh.2013.02.001>
- Carrizo, D., y Alfaro, A. (2018). Método de aseguramiento de la calidad en una metodología de desarrollo de software: un enfoque práctico. *Ingeniare. Revista Chilena de Ingeniería*, 26(1), 114-129. <http://dx.doi.org/10.4067/S0718-33052018000100114>
- Dalal, S., Solanki, K., Sudhir, & Diksha. (2019). Exploring the essentials and principles of software development. *International Journal of Advanced Trends in Computer Science and Engineering*, 8(6) 3504-3510. <https://doi.org/10.30534/ijatcse/2019/129862019>
- Delgado Olivera, L. de la C., y Diaz Alonso, L. M. (2021). Modelos de desarrollo de Software. *Revista Cubana de Ciencias Informáticas*. 15(1), 37-51. <https://bit.ly/3QW2oZU>
- Galli, T., Chiclana, F., & Siewe, F. (2021). On the use of quality models to address distinct. *Applied System Innovation*, 4(3), 41. <https://doi.org/10.3390/asi4030041>

- García Peñalvo, F. (2018). Ingeniería del Software. Ingeniería del Software y Gobierno de Tecnologías de la Información. 277-388. Salamanca: Universidad de Salamanca. <https://repositorio.grial.eu/bitstream/grial/1228/4/07-rep.pdf>
- Génova, G., Astudillo, H., & Fraga, A. (2016). The scientometric bubble considered harmful. *Science and Engineering Ethics*, (22), 227-235. <https://doi.org/10.1007/s11948-015-9632-6>
- Gómez Palomo, S., y Moraleda Gil, E. (2020). Aproximación a la ingeniería del software. Madrid: Editorial Universitaria Ramon Areces.
- Organización Internacional de Normalización (ISO) y Comisión Electrotécnica Internacional (IEC). (2017). ISO/IEC TS 25011:2017 Information technology — Systems and software Quality Requirements and Evaluation (SQuaRE) — Service quality models. Ginebra, Suiza. <https://www.iso.org/standard/35735.html>
- Jaimes Hernández, A. F. (2020). *Análisis bibliométrico desde las ciencias de la información y su relación con el desarrollo sostenible y la industria 4.0 2015-2020* [Tesis de Grado, Universidad de La Salle, Colombia]. [https://ciencia.lasalle.edu.co/sistemas\\_informacion\\_documentacion/756/](https://ciencia.lasalle.edu.co/sistemas_informacion_documentacion/756/)
- Khan, K., & Saleh, M. (2021). Understanding the impact of emotions on the Quality of Software Artifacts. *IEEE Access*, 9, 110194 - 110208. <https://doi.org/10.1109/ACCESS.2021.3102663>
- Nees Jan van, E., y Waltman, L. (2023). VOSviewer Manual. Manual de usuario, Amsterdam. [https://www.vosviewer.com/documentation/Manual\\_VOSviewer\\_1.6.19.pdf](https://www.vosviewer.com/documentation/Manual_VOSviewer_1.6.19.pdf)
- OMPI, Organización Mundial de la Propiedad Intelectual. (22 de septiembre de 2022). Índice Mundial de Innovación 2022. [https://www.wipo.int/pressroom/es/articles/2022/article\\_0011.html](https://www.wipo.int/pressroom/es/articles/2022/article_0011.html)
- Pantaleo, G., y Rinaudo, L. (2016). Ingeniería de Software. Buenos Aires: Alfaomega Grupo Editor.
- Peláez Valencia, L., Hurtado Gil, S., Toro Lazo, A., y Arias Vargas, L. (2018). Aseguramiento de la Calidad del Software: Acercamiento al Marco Teórico y al Estado de la Cuestión desde la Disciplina de la Ingeniería de Software. En Tecnológico de Antioquia - Institución Universitaria (Ed.), Investigación e Innovación en Ingeniería de Software (págs. 19-30). Publicar T – Sello Editorial TdeA. <https://bit.ly/47OIBkR>
- Piattini Velthuis, M. (2016). Evolución de la Ingeniería del Software y la formación de profesionales. *Bit & Byte*, (4), 15-17. <https://bit.ly/3EdO6w2>
- Pressman, R. (2010). Ingeniería del Software. Un enfoque práctico. México: McGrawHill.
- Research.com. (2023). Best Journals - Computer Science/IEEE Access. <https://research.com/journal/ieee-access>
- Research.com. (2023). Best Journals - Computer Science/IEEE Transactions on Software Engineering. <https://research.com/journal/ieee-transactions-on-software-engineering-1>
- Ruiz de Mendarozqueta, A., Bustos, F., y Colla, P. (2022). Relación entre las prácticas maduras de ingeniería de software y las prácticas de agilidad. *Electronic Journal of SADIO*, 21(2), 79-109. <https://ojs.sadio.org.ar/index.php/EJS/article/view/222>
- Sánchez Jaya, V., Robayo Jácome, D., y López Sevilla, G. (2019). Apuntes teóricos sobre modelos de evaluación de calidad en procesos de desarrollo de software para personas no videntes [Ponencia]. IV Congreso Internacional Ciencia Sociedad e Investigación Universitaria, Ambato, Ecuador. <https://repositorio.pucesa.edu.ec/handle/123456789/2709>

- Solano López, E., Castellanos Quintero, S. J., López Rodríguez del Rey, M. M., y Hernández Fernández, J. I. (2019). La bibliometría: una herramienta eficaz para evaluar la actividad científica post-graduada. *Revista Electrónica de las Ciencias Médicas en Cienfuegos*, 7(4), 59-62. <https://bit.ly/3slxGyS>
- Sommerville, I. (2011). *Ingeniería de Software*. México: Pearson.
- Torres Fernández, P. A. (2016). Acerca de los enfoques cuantitativo y cualitativo en la investigación educativa cubana actual. *Atenas*, 2(34), 1-15. <https://bit.ly/3sse78a>
- Valdovinos, G. (2023). El futuro del desarrollo de software: tendencias emergentes y oportunidades de crecimiento en 2023. <https://es.linkedin.com/pulse/el-futuro-del-desarrollo-de-software-tendencias-y-en-2023-valdovinos>
- Van Eck, L., Waltman, L., Dekker, R., & Van Den Berg, J. (2010). A comparison of two techniques for bibliometric mapping: Multidimensional. *Journal of the American Society for Information Science and Technology*, 2405-2416. <https://doi.org/10.48550/arXiv.1003.2551>
- Velásquez, M., Vahos Montoya, J., Gómez Adasme, M., Restrepo Zapata, J., Pino, A., y Londoño Marín, S. (2019). Una revisión comparativa de la literatura acerca de metodologías tradicionales y modernas de desarrollo de software. *Revista Cintex*, 24(2), 13-23. <https://doi.org/10.33131/24222208.334>
- Veloz Segura, E. (2022). Componentes de calidad software y su utilización en aplicaciones web. *Ciencia Latina Revista Científica Multidisciplinar*, 3193-3204. [https://doi.org/10.37811/cl\\_rcm.v6i3.2456](https://doi.org/10.37811/cl_rcm.v6i3.2456)
- Winters, T., Manshreck, T., y Wright, H. (2022). *Ingeniería de software en Google: Lecciones sobre programación aprendidas a lo largo del tiempo*. Marcombo.