

# PaperSeek: un buscador de publicaciones científicas en bases de datos electrónicas

Uriel Paredes<sup>1</sup>, Eder dos Santos<sup>1</sup>[0000-0001-6729-0303] Sandra Casas<sup>1</sup>[0000-0002-8289-6132]

<sup>1</sup> Universidad Nacional de la Patagonia Austral, Río Gallegos 9400, Argentina  
paredesuri@gmail.com, {esantos, sicasas}@unpa.uarg.edu.ar

**Resumen.** Este trabajo presenta los resultados del desarrollo de una herramienta para facilitar la realización de revisiones sistemáticas de literatura científica en múltiples bases de datos. La revisión sistemática se destaca como una metodología que permite identificar, evaluar e interpretar las investigaciones y estudios disponibles sobre un tema específico. Se describe a continuación la propuesta desarrollada con los documentos que fueron elaborados, desde la formulación de visión y objetivos hasta el modelo de diseño y pruebas de una aplicación llamada Proyecto PaperSeek. La aplicación desarrollada permite realizar búsquedas simultáneas en las bases de datos científicas ACM Digital Library, ScienceDirect, IEEE Xplore y SpringerLink, con funcionalidades que incluyen búsqueda simple y búsqueda por intervalo de tiempo. Se discute el diseño del sistema, la gestión de configuración utilizando GitHub, y se presentan casos de prueba que validan el correcto funcionamiento del sistema desarrollado. Al final de este documento se encuentran las conclusiones obtenidas al finalizar el trabajo de desarrollo.

**Palabras Clave:** Búsqueda y Referencia, Ingeniería de Software, Desarrollo de Software, Bases de Datos Científicas.

## 1 Introducción

El proceso de investigación suele comenzar con una revisión de literatura. Dicha revisión sistemática permite identificar, evaluar e interpretar todas las investigaciones y estudios disponibles que son relevantes para alguna pregunta de investigación particular, un área temática o algún fenómeno de interés [1][2]. Como resultado, una revisión permite llevar a cabo un resumen de la evidencia que existe en lo que respecta a un tratamiento o tecnología; identificar huecos en la investigación existente para sugerir áreas en las que continuar investigando; o para proporcionar un marco para posicionar nuevas actividades de investigación correctamente. También se pueden usar las revisiones sistemáticas para examinar la medida en que la evidencia empírica apoya o contradice las hipótesis teóricas, o incluso como ayuda para crear nuevas hipótesis. Para que una revisión sea rigurosa, imparcial, transparente y replicable, se adopta un protocolo de revisión sistemática, a través del cual se especifican los métodos que se utilizarán para llevar a cabo la revisión. Los componentes del protocolo incluyen todos los elementos de la revisión, tales como: antecedentes, preguntas de investiga-

ción, estrategia que se utilizará para buscar, criterios y procedimientos de selección de estudios, listas de control y procedimientos de evaluación de la calidad de los estudios, estrategia para la extracción de datos, síntesis de los datos extraídos, y el cronograma del proyecto. La parte más importante del protocolo es la formulación de preguntas de investigación. Según lo planteado por [1], una pregunta de investigación debe ser significativa e importante para profesionales de la industria e investigadores, asimismo que pueda llevar a cambios en las prácticas actuales o a una mayor confianza en éstas, y que puede identificar discrepancias entre creencias comunes y la realidad.

Una vez definido el protocolo, se genera inicialmente una estrategia de búsqueda. Las búsquedas iniciales para estudios primarios. Uno de los métodos principales para recopilar información es la búsqueda en bases de datos electrónicas de publicaciones científicas. Dichas bases de datos ofrecen acceso a un amplio acervo de literatura científica que puede ser utilizada como referencia en un proyecto de investigación, y por lo general proveen mecanismos de búsqueda que permiten acotar los resultados de búsqueda adecuados al marco de trabajo y los objetivos de investigación. Estos repositorios suelen contener publicaciones organizadas o especializadas por disciplinas.

A pesar de la utilidad y versatilidad que ofrecen las bases de datos, realizar búsquedas en las mismas trae consigo algunos inconvenientes. En primer lugar, no todas las bases de datos tienen la misma cantidad de artículos disponibles. Un artículo que se muestre en una base de datos podría no encontrarse en otra, lo que hace necesario revisar múltiples bases de datos para poder encontrar la mayor cantidad de información. Tener que buscar en varias bases de datos al mismo tiempo aumenta significativamente la cantidad de trabajo a realizar, ya que se tendrá que realizar una misma búsqueda varias veces por cada base de datos que se utilice. Además, el funcionamiento de cada base de datos puede tener diferencias significativas, como pueden ser los parámetros admitidos por cada una al momento de realizar una búsqueda avanzada. Para abordar este desafío, en este trabajo se presenta el desarrollo de una herramienta que permite realizar una búsqueda integrada en distintas bases de datos científicas.

El objetivo general de este trabajo es presentar el desarrollo de PaperSeek, un buscador unificado de publicaciones científicas. Se describe el método de desarrollo utilizado, los artefactos elaborados, las herramientas que han soportado el proceso y los resultados obtenidos a partir de distintos casos de prueba.

## 2 Métodos

La obtención de un software con calidad implica la utilización de metodologías o procedimientos estándares para el análisis, diseño, programación y prueba del software que permitan uniformar la filosofía de trabajo, con el fin de lograr una mayor confiabilidad, mantenimiento y facilidad de prueba, a la vez que eleven la productividad, tanto para la labor de desarrollo como para el control de la calidad del software.

Se ha adoptado PSI<sup>1</sup>, un marco de trabajo basado en el Proceso Unificado para el Desarrollo de Software [5]. PSI se caracteriza por estar dirigido por Casos de Uso, cen-

---

1 <https://www.uarg.unpa.edu.ar/psi/>

trado en la arquitectura y por ser iterativo, incremental y extensible, que puede ser adaptado por a organizaciones o procesos específicos.

Para ese desarrollo se definió inicialmente la visión del sistema. Se elaboró posteriormente una especificación de requerimientos y un plan de gestión de configuración. El desarrollo además incluyó la creación de un modelo de diseño, y para poder comprobar el correcto funcionamiento de la aplicación se utilizaron dos casos de prueba parametrizados: el primer caso de prueba se realizó a partir de una cadena de búsqueda, y el segundo caso incorporó un intervalo de tiempo en años a la cadena original. Finalmente, también se llevó a cabo la elaboración de un manual de instalación.

El desarrollo involucró la selección de distintas bases de datos [4]. Se eligieron las siguientes bases de datos: IEEE Xplore, ACM Digital Library (DL), ScienceDirect y Springer. IEEE Xplore y ACM DL se eligieron por su cobertura de actas de congresos y artículos de revistas relevantes publicados por IEEE o ACM. Los repositorios ScienceDirect y SpringerLink se seleccionaron por su representación de artículos de las principales revistas de ingeniería de software de todo el mundo. Cabe mencionar que todos los repositorios seleccionados incluyen estudios revisados por pares.

### **3 Propuesta Desarrollada**

#### **3.1 Visión del Sistema**

En el documento de visión se establecieron los objetivos y el alcance del Proyecto PaperSeek en relación con las dificultades para realizar revisiones sistemáticas en múltiples bases de datos. Teniendo en cuenta esta problemática, se decidió desarrollar una aplicación que permita realizar búsquedas en múltiples bases de datos científicas al mismo tiempo, con el objetivo de reducir el tiempo necesario para llevar a cabo una revisión sistemática de la literatura científica. Las bases de datos que se utilizarán son ACM Digital Library, ScienceDirect, IEEE Xplore y SpringerLink.

#### **3.2 Especificación de Requerimientos**

En la especificación de requerimientos se determinaron los principales aspectos a considerar para el desarrollo de las funcionalidades de la aplicación. Estos requerimientos incluyen la capacidad de realizar una búsqueda simple, seleccionar las bases de datos en las que se llevará a cabo la búsqueda y poder especificar un intervalo de tiempo determinado por años para la búsqueda.

También se incluyó un modelo de casos de uso, centrado en el caso de uso “Búsqueda Simple” como la principal funcionalidad de la aplicación. Además, se listaron requerimientos no funcionales en cuanto a usabilidad, confiabilidad, seguridad y eficiencia. Se destaca, además, una captura de pantalla de la aplicación para mostrar el diseño de la interfaz gráfica.

### 3.3 Plan de Gestión de Configuración

La gestión de configuración se llevó a cabo utilizando un repositorio en GitHub <sup>2</sup>. El equipo realizó reuniones virtuales a través de Google Meet para revisar y acordar los cambios necesarios a implementar en el código.

Basado en lo acordado por el equipo, los cambios serían implementados y subidos a la rama correspondiente de GitHub utilizando una extensión que permite conectar repositorios de GitHub con el IDE utilizado para el desarrollo: Visual Studio Code.

Durante el transcurso de , se brindó capacitación y recursos para poder aprovechar las utilidades de GitHub y llevar a cabo una gestión de configuración exitosa.

### 3.4 Modelo de Diseño

El modelo de diseño expone la estructura utilizada para la aplicación desarrollada. PaperSeek es una aplicación web modular orientada a objetos, desarrollada en lenguajes HTML5, PHP y Javascript, soportada por el *framework* Bootstrap<sup>3</sup>. Principalmente, cuenta con dos clases: ControlGUIForm y Parser. ControlGUIForm es un controlador que gestiona aspectos de la interfaz gráfica relacionados con el formulario de búsqueda de la aplicación, mostrando distintos resultados y secciones de acuerdo con las entradas ingresadas por el usuario que realiza la búsqueda. La clase Parser es una superclase diseñada para generar los enlaces URL correspondientes a las búsquedas en cada base de datos utilizada. La implementación de sus métodos varía según la subclase, ya que cada base de datos utiliza un formato de URL diferente para sus búsquedas. Las subclases de Parser son: ParserACM, ParserScienceDirect, ParserIEEEExplore y ParserSpringerLink. En el documento también se incluye un diagrama de clases de la aplicación desarrollada.

### 3.5 Casos de Prueba

Para documentar las pruebas realizadas, se elaboraron dos documentos detallando dos casos de prueba: uno para la búsqueda simple y otro para la búsqueda con intervalo de tiempo. En ambos documentos se especifican las condiciones y entradas utilizadas durante las pruebas.

En el primer caso de prueba se utilizó la cadena de búsqueda “API Management”, con las cuatro bases de datos seleccionadas. Se dejaron vacíos los campos correspondientes a un intervalo de búsqueda, esperando que el resultado sea una búsqueda simple que incluye resultados de cualquier año de publicación. Para el segundo caso de prueba se utilizaron condiciones similares, con la misma cadena de búsqueda y bases de datos seleccionadas, pero esta vez se rellenaron los campos correspondientes a un intervalo de búsqueda con los años 2019 y 2024. El resultado esperado consistía en la búsqueda de la cadena “API Management” en las cuatro bases de datos, pero limitándose solo a aquellos resultados publicados entre los años 2019 y 2024. En ambos casos, los resultados de las pruebas fueron satisfactorios, mostrando un correcto funcionamiento del sistema según las condiciones establecidas.

---

2 <https://www.github.com>

3 <https://getbootstrap.com>

### 3.6 Manual de Instalación

El último documento elaborado es un manual de instalación, que detalla el proceso de instalación de la aplicación. Es necesario utilizar una aplicación que permita configurar un servidor web, como XAMPP. A continuación, se instala la aplicación en la carpeta correspondiente según la aplicación utilizada (htdocs para XAMPP). Finalmente, solo es necesario iniciar el servidor y abrir un navegador web para ingresar a <http://localhost/buscador/buscador.php> y poder comenzar a utilizar la aplicación.

## 4 Conclusiones

Con el trabajo realizado se logró llevar a cabo el desarrollo de una herramienta que permite realizar una búsqueda en cuatro bases de datos simultáneamente. Además, también es posible establecer un intervalo de tiempo por años para limitar la cantidad de artículos que se muestran. Las funcionalidades de esta aplicación resultan de gran utilidad para agilizar los procesos de búsqueda en bases de datos, lo que es útil para la revisión sistemática de bibliografía científica. Solo se necesita ingresar los parámetros de búsqueda una única vez, simplificando el proceso y evitando posibles errores que pudieran cometerse realizando las búsquedas de manera manual. Esto resulta muy beneficioso para la investigación y por ello una herramienta de este tipo tiene una gran relevancia para la comunidad científica. Como trabajo futuro se plantea expandir las capacidades de la herramienta, principalmente a través de la adopción de más bases de datos e incluyendo funcionalidades de búsqueda avanzada tales como los operadores binarios lógicos y una especificación más detallada de distintos metadatos.

## Referencias

- [1] Kitchenham, B., Brereton, O. P., Budgen, D., Turner, M., Bailey, J., & Linkman, S. (2009). Systematic literature reviews in software engineering – A systematic literature review. *Information and Software Technology*, 51(1), 7–15. <https://doi.org/10.1016/j.infsof.2008.09.009>.
- [2] Petersen, K., Feldt, R., Mujtaba, S., & Mattsson, M. (2008). Systematic mapping studies in software engineering. *Electronic Workshops in Computing*. <https://doi.org/10.14236/ewic/ease2008.8>
- [3] Kitchenham, B. (2004). Procedures for performing systematic reviews. ResearchGate. [https://www.researchgate.net/publication/228756057\\_Procedures\\_for\\_Performing\\_Systematic\\_Reviews](https://www.researchgate.net/publication/228756057_Procedures_for_Performing_Systematic_Reviews)
- [4] Dyba, T., Kitchenham, B., & Jorgensen, M. (2005). Evidence-based software engineering for practitioners. *IEEE Software*, 22(1), 58–65. <https://doi.org/10.1109/ms.2005.6>
- [5] Booch, G., Rumbaugh, J., & Jacobson, I. (2000). UML: el proceso unificado de desarrollo de software. Addison-Wesley.