Received: 10 August 2025 | Revised: 5 December 2025 | Accepted: 18 December 2025 | Published online: 29 December 2025

RESEARCH ARTICLE

Artificial Intelligence and Applications
2025, Vol. 00(00) 1–13
DOI: 10.47852/bonviewAIA52027141

BON VIEW PUBLISHING

Development of a Web Workflow Tool That Facilitates and Optimizes the Process of Preparing Scientific Articles

Lucrecia Llerena[1,*], Nancy Rodríguez[1], and Kelvin Estrada[1]

[1] *Software Engineering Program, Quevedo State University, Ecuador*

Abstract: Writing scientific articles is a complex process for novice researchers, who often face difficulties related to content structuring, the proper use of academic language, and reference management. This research presents the development and validation of a web tool called RedactFlow, designed to support the preparation of scientific manuscripts through functionalities that integrate user-centered design principles and artificial intelligence (AI) techniques. The methodology used was non-experimental with a cross-sectional design, and included a systematic literature review to provide theoretically support for the design, as well as the use of the ISO 9241-210:2019 standard for the construction of user-experience-oriented interfaces. The system was developed using an adapted version of the agile SCRUM methodology, and incorporates modules for guided content generation, grammatical correction, real-time suggestions, reference management, and export to standardized academic formats. The validation of the system was carried out through functional, performance, and usability tests, including comparisons between tool-generated texts and expert-revised versions. The results showed significant improvements in the coherence, clarity, and formal structure of the texts, as well as a satisfactory user experience, reflected in an average score of 76.75 on the System Usability Scale (SUS). In addition, the system demonstrated stable behavior under high-demand conditions, maintaining optimal response times. Overall, the findings support the conclusion that RedactFlow is an innovative and effective proposal to support scientific writing, especially in its initial stages, by providing a structured workflow that contributes to strengthening academic writing skills in both training and research contexts.

Keywords: scientific writing, artificial intelligence (AI), web tools, automation, academic publishing, usability, AI-assisted writing

1. Introduction

Scientific communication is key to the progress of knowledge, as it facilitates the dissemination of discoveries. However, the growing information overload, caused by the exponential increase in data, represents a major challenge for current research by Arnold et al. [1]. This phenomenon especially affects researchers working in fields with abundant scientific literature, increasing the need for technological solutions that optimize the planning, writing and management of scientific articles [2].

Currently, researchers face multiple challenges when preparing scientific manuscripts, including fragmented tools, a lack of automation in formatting processes, and manual reference management. Recent studies show that formatting and reformatting manuscripts takes an average of 14 h per article (≈52 h/year per researcher), with 91% of authors spending >4 h and 65% spending >8 h; Furthermore, these demands delay resubmission by >2 weeks in most cases and by >3 months in around 20% of manuscripts [3]. This issue highlights the need for a unified environment that optimizes the stages of writing, reviewing, and formatting scientific articles.

Added to this need is the growing complexity and volume of scientific literature has generated an environment capable of overwhelming beginner authors and making it difficult to plan, write and manage academic texts. In response, there has been a growing interest in smart technological tools that support writing processes from a structured and efficient perspective [2]. In this context, Artificial Intelligence (AI), particularly Natural Language Processing (NLP), has shown remarkable potential to improve scientific writing. NLP techniques make it possible to automatically detect grammatical errors automatically, improve the coherence of texts and offer personalized recommendations that favor autonomous learning [4]. Tools such as ChatGPT have shown their usefulness in educational environments; however, most are not adapted to the pedagogical demands of academic writing or the specific workflow it requires [5, 6].

Various studies indicate that the writing of scientific articles presents frequent problems, such as disorganization, inconsistencies and grammatical errors, especially among those who are new to this process [7–9]. Therefore, there is a growing need for digital tools that offer a structured, guided, and interactive workflow from the earliest stages of the writing process.

Although there are widely used collaborative tools such as Overleaf, Authorea, and SciFlow, these focus mainly on shared editing and document generation in LaTeX, without offering intelligent assistance or alignment with local editorial standards. RedactFlow therefore emerges as an innovative alternative that integrates task automation, structural validation following the IMRaD format, and intelligent assistance through AI into a single environment, addressing the real needs of researchers in Spanish-speaking academic contexts.

In this context, the present research focuses on the development of RedactFlow, a web-based workflow system designed to support novice researchers in writing scientific papers. This tool, based on AI, aims to facilitate the first phases of the writing process through an intuitive interface and functionalities that promote the organization and clarity of ideas.

Among its most outstanding features is the automatic generation of writing schemes for each section of the article, as well as an

***Corresponding author:** Lucrecia Llerena, Software Engineering Program, Quevedo State University, Ecuador. Email: lllerena@uteq.edu.ec

intelligent assistant that guides the user in structuring and writing content. RedactFlow also allows the user to export an initial draft in PDF, Word, and LaTeX formats, reducing the difficulty of starting from scratch and improving document customization. In addition, it incorporates specific templates according to the type of article, which contributes to the standardization of the format and improves adherence to editorial guidelines.

Unlike other existing tools such as Grammarly, mainly focused on grammatical and stylistic correction [10, 11], ChatGPT, used to generate textual drafts based on general instructions [12], or Paperpal, aimed at improving academic language through automatic suggestions and correction of errors in scientific texts [4]. RedactFlow integrates structured content generation into a unified environment, intelligent assistance in academic writing, contextualized review and export to multiple scientific formats (PDF, Word, LaTeX), offering a complete and guided workflow, specially designed to support novice researchers during the initial stages of the editorial process. It also integrates with Zotero to facilitate automatic management of bibliographic references. These functionalities seek not only to simplify the process, but also to progressively strengthen the user's academic writing skills.

The development of RedactFlow was carried out following the agile SCRUM methodology, which allowed an iterative, controlled implementation focused on the real needs of users. Its evaluation was carried out by applying principles of Human–Computer Interaction (HCI), post-test surveys and the System Usability Scale (SUS). Likewise, drafts generated by the tool were compared with peer-reviewed versions, allowing their impact on improving textual quality and user experience to be measured.

In terms of contribution, this study presents an original solution that combines NLP algorithms, academic structuring heuristics, and interactive review modules, integrated into an accessible and usable system for the emerging scientific community. The main contribution lies in the proposal and validation of RedactFlow, a web-based system that unifies these components into a coherent workflow for academic writing. Unlike other existing tools, RedactFlow combines automation, pedagogical assistance, and editorial standardization in a unified environment. Its originality lies in the empirical validation of its effectiveness in terms of textual quality, usability, and performance, demonstrating its potential as both a formative and technical resource for novice researchers in the early stages of scientific writing.

The main innovation of RedactFlow lies in the integration of intelligent modules based on NLP (AI-GPT), automatic reference management using the Zotero API, and a guided pedagogical system that accompanies the user in the construction of the article under the IMRaD scheme. This combination of automation, assistance, and guided learning sets RedactFlow apart from traditional scientific writing platforms.

This article is organized into several sections. Section 2 presents the theoretical foundation that supports the development of the proposed tool. Section 3 describes in detail the process of designing and implementing the web system, as well as the tools used for its validation. Section 4 presents the results obtained from the tests of operation, quality, performance and usability. Finally, Section 5 discusses the findings, and Section 6 presents the conclusion of the study, along with recommendations for future research and potential functional improvements.

2. Related Work

In recent years, there has been a significant increase in the application of AI-based tools to support scientific writing, especially among novice researchers. Various studies have analyzed their impact on aspects such as textual organization, grammatical correctness, and generation of academic content.

One of the most prominent approaches is the use of language models such as ChatGPT. Singh et al. [13] explored its usefulness in organizing ideas, writing drafts, and revising texts, highlighting its future potential in scientific publications, although they stressed the need for human oversight due to potential errors or biases. Along the same lines, Khaleel et al. [14], identified that ChatGPT can speed up the writing process, help structure ideas, and improve style, but warned of risks such as data invention or plagiarism.

Kacena et al. [12] compared three approaches in writing review articles (human, AI, and combination), concluding that AI reduces writing time, but introduces errors, especially in references. For his part, Chetwynd [15] emphasized the ethical dilemmas associated with the use of AI in scientific writing, advocating for transparency and the application of ethical guidelines such as those of the ICMJE and COPE to avoid the generation of low-quality science.

In the field of education, Khabib [16] analyzed the use of AI-based writing assistants among teachers, revealing improvements in efficiency and quality of texts. Also, Giglio and Costa [4] highlighted that these tools can be especially useful for non-native English speakers, by facilitating clarity and grammatical correctness, increasing the probability of acceptance in scientific journals.

From a comprehensive educational perspective, Kusmanto et al. [17] conducted a systematic review on the implementation of AI in the teaching of scientific writing, covering publications between 2020 and 2024. Their study identified nine key dimensions of AI application in this area: real-time feedback, development of writing skills and adaptive learning, productivity improvement, reference resources, textual quality analysis, strengthening critical thinking, personalized learning, autonomous study, and collaboration in the writing process. The authors highlight that, although AI brings substantial benefits, its use must be guided by ethical principles and clear regulations to avoid adverse effects on the educational process.

These works coincide in highlighting the value of AI as a support tool in academic writing, although they also warn about its current limitations. Shortcomings are identified in essential functionalities for scientific production, such as structuring by sections, integration with bibliographic managers, or compatibility with specialized formats such as LaTeX.

Faced with these limitations, there is a need for more comprehensive solutions, capable of combining language assistance with structural and technical support. In response to this challenge, RedactFlow is proposed, a web tool that incorporates NLP techniques to facilitate a complete academic workflow. Its design is based on consolidated theoretical approaches and the best practices of scientific writing, as will be detailed in the subsequent methodological sections.

In recent years, various studies have explored the application of AI in scientific writing and academic publishing support systems [18, 19]. These studies agree that AI contributes to improving textual coherence and manuscript organization, but limitations remain in integrating intelligent functions with complete editorial workflows, reinforcing the relevance of the approach proposed by RedactFlow.

3. Research Methodology

The methodology adopted in this study corresponds to a non-experimental approach with a cross-sectional design and was structured into four main stages: first, a systematic review was conducted to establish the theoretical and technical foundations of the proposal; secondly, the methodological guidelines were defined based on the specific objectives of the research; thirdly, the proposed software tool was developed through an iterative process; and finally, a comprehensive evaluation of the system was carried out to analyze its performance and usability.

Table 1
Databases and search fields used

Database	Search fields used
IEEE Xplore	Title, Author Keywords, Abstract
ACM Digital Library	Title, Author Keywords, Abstract
ScienceDirect	Title, Keywords, Abstract
SpringerLink	Title, Author Keywords

3.1. Systematic review (SMS)

To support the development of the proposed solution, a Systematic Mapping Study (SMS) was conducted aimed at identifying current techniques and methodologies applied to the writing of scientific articles. This review answered the following research question: What writing techniques and methodologies are currently used to improve consistency, clarity, and accuracy in scientific research writing? The process began with the definition of the search query, which was designed based on key terms related to scientific writing and writing methodologies. The final search string used was: ((“methodologies” OR “strategies” OR “approaches”) AND (“scientific writing” OR “academic writing” OR “article preparation”) AND (“scientific articles” OR “academic publications” OR “papers”)). The search was carried out in four recognized databases, in which specific filters were configured to improve the accuracy of the results. These results can be found in Table 1. In addition, criteria were established to ensure the quality and relevance of the articles analyzed. These criteria are presented in Table 2.

The result of the filtering and selection process is presented in Table 3.

The analysis of the primary studies enabled the identification of a set of recurrent techniques focused on improving key aspects such as clarity, structural coherence, terminological adequacy, and conceptual precision in scientific writing. These findings served as methodological basis for the design of the proposed system, ensuring its alignment with the real needs of the academic writing process.

3.2. Design of user interfaces

The design of the user interfaces of the web tool was developed under a User-Centered Design (UCD) approach, in accordance with the guidelines established by the international standard ISO 9241-210:2019. This standard promotes an iterative process that comprehensively considers the needs, expectations, and context of use of end users, with the aim of improving the usability, accessibility, and efficiency of interactive systems [20].

The global architecture supporting this interface design is described in detail in Section 4.3.1 (System Architecture), where the component structure and communication flow between the frontend, backend, and external API services are detailed.

The design process began with an initial phase of information collection and analysis, through which the main challenges faced by novice researchers during scientific writing were identified. The central purpose was to ensure that the resulting user interface aligned with these real requirements, facilitating an intuitive, coherent, and pedagogical experience.

Table 2
Inclusion and exclusion criteria

Inclusion criteria	Exclusion criteria
Articles written in English	Works in Spanish
Published between 2019 and 2025	These or other non-refereed work
Studies describing scientific writing techniques or methodologies	Articles not directly related to scientific writing processes

Table 3
Database search process results

N°	Database	Results
1	IEEE Xplore	2
2	ACM Digital Library	2
3	ScienceDirect	11
4	SpringerLink	91

Following the conceptual framework of ISO 9241-210, the process was articulated in three fundamental stages that guided the design in a structured way:

1) Understanding and specification of the context of use: The operating conditions of the system were analyzed, as well as the profiles, capacities, and specific needs of the users. This analysis was based on both a systematic literature review and semi-structured interviews with potential users, which enabled the construction of an accurate representation of the interaction environment.
2) Specification of user requirements: From the previous diagnosis, functional requirements (associated with key system functionalities, such as assistance in article structuring, automated feedback, and reference management) and non-functional requirements (including accessibility, cross-platform compatibility, performance, and security) were established. This specification ensured a direct alignment between the proposed functionalities and the detected demands.
3) Generation of design solutions: Interactive prototypes were developed using the Figma tool, prioritizing clear, hierarchical, and task-focused visual architecture. The prototypes were evaluated internally and subsequently adjusted based on feedback from real users, allowing for an iterative improvement of the design in terms of usability and efficiency.

These three stages fed into a continuous cycle of improvement, which allowed the design of the interface to be progressively refined based on the learnings derived from direct interaction with users. This approach ensured the creation of a technological tool that is not only functional, but also adapted to the real conditions of use, particularly in educational contexts and initiation into scientific writing.

3.2.1. Understanding and specifying the context of use

This stage focused on analyzing the operating environment of the system, as well as identifying the specific needs, expectations, and constraints of end-users – novice researchers in the field of scientific writing. To this end, two complementary sources of information were used: (i) a systematic bibliographic review (described in Section 3.1), which enabled the recognition of key methodological and theoretical approaches in academic writing, and (ii) semi-structured interviews conducted with five potential users, who provided empirical evidence on their experiences, difficulties, and requirements in relation to technological tools to support writing.

The interviews were designed to explore critical aspects of the writing process, such as the most frequent obstacles in the preparation of scientific articles, familiarity with digital assistance platforms, and expectations regarding a tool that optimizes this process. Each session lasted approximately 15 min, was recorded with the informed consent of the participants, transcribed in its entirety, and then subjected to a qualitative analysis to identify common patterns and recurring needs.

Table 4
Matrix of relationship between identified problems and proposed functional solutions

Problem identified	Proposed functional solution
Difficulty structuring content	Structure wizard with predefined sections
Ignorance of academic conventions	Interactive guides integrated into the system
Poor planning of the writing process	Organization module with timelines and tasks
Absence of feedback	Automatic improvement suggestions in real time

The triangulation between the documentary findings and the testimonies collected made possible the identification of the main challenges faced by novice researchers, among which the following stand out:

1) Difficulties in coherently structuring scientific articles.
2) Poor command of style rules and formal criteria of textual organization.
3) Limited experience in the use of digital tools that support or automate tasks in the writing process.
4) Uncertainty in the planning of the content and in the argumentative cohesion of their writings.

As a result of this contextual diagnosis, a matrix was constructed that links each identified problem with a corresponding functional solution within the design of the tool, presented in Table 4.

3.2.2. Specifying user requirements

Based on a detailed understanding of the context of use, the functional and non-functional requirements of the web system were systematically defined, with the aim of ensuring that the tool responds accurately and effectively to the needs detected in end users.

1) Functional requirements: Key functionalities aimed at facilitating the scientific writing process were identified, particularly for researchers with limited experience. Among the prioritized capabilities are the structural organization of articles through dynamic templates, automated assistance in the writing of specific sections, real-time feedback based on NLP, and the management of bibliographic references through integration with external tools.
2) Non-functional requirements: These were considered essential quality attributes to ensure a robust and satisfactory user experience. These requirements include: (i) accessibility, ensuring that the system can be used by people with different levels of technical ability; (ii) security, through authentication, encryption, and data protection mechanisms; (iii) performance, focused on minimum response times and stability in the face of high user concurrency; and (iv) usability, ensuring an intuitive, coherent, and adaptable interface to various devices and access contexts.

This specification served as the basis to guide the architectural design of the system and the prioritization of functionalities during the subsequent stages of development and implementation.

3.2.3. Generation of design solutions

Once the system requirements were defined, the design phase of the graphical interface began by using the Figma tool, selected for its versatility in the creation of high-fidelity interactive prototypes. This stage focused on ensuring a fluid, intuitive, and visually coherent user experience, prioritizing clarity in the arrangement of elements, the readability of the contents, and the ease of navigation.

In the first instance, preliminary versions of the interfaces were developed, with the aim of evaluating the general structure, visual hierarchy, and interaction flow of the platform. These first designs were subjected to an internal review process that enabled the identification of areas for improvement in terms of visual organization and accessibility.

Subsequently, advanced prototypes were developed that incorporated interactive elements and simulated key functionalities of the system. These prototypes were used in test sessions with users who were representatives of the target audience, who provided qualitative feedback on their user experience, identifying positive aspects, perceived difficulties, and suggestions for improvement.

This iterative approach, focused on continuous feedback and early validation, enabled the progressive refinement of the interface, ensuring its alignment with the real expectations and needs of users. As a result, a functionally robust and visually accessible design was consolidated, which favors the acceptance, adoption, and effectiveness of the tool in real contexts of academic use.

3.3. Development of the system under adapted agile SCRUM methodology

After the definition of the theoretical and technical foundations, the development of the RedactFlow tool was structured using the agile SCRUM methodology, which was adapted to be executed by a single developer. Although SCRUM was originally conceived for collaborative teams with differentiated roles, in this case, all the functions (Product Owner, Scrum Master, and Development Team) were assumed by a single person. This particularity required meticulous planning, disciplined self-management, and rigorous control of the progress of the project.

In this framework, the role of Product Owner was replaced by an initial planning phase, during which the general objectives of the system were defined and the essential functionalities to be developed were prioritized. The role of Scrum Master was internalized as a personal mechanism for monitoring and reflecting on progress. Furthermore, the technical execution included both the development of the frontend (in Angular) and the backend (in Spring Boot), along with the database administration in MySQL and the integration of external services based on AI.

3.3.1. Sprint structure and feature prioritization

The development process was organized into six monthly sprints, covering a total period of 6 months. The management of the backlog was carried out through a structured spreadsheet, where the functionalities were listed and hierarchized according to their technical criticality and dependence between modules. This prioritization ensured a minimal operational base before incorporating advanced features.

During the course of the sprints, new needs were identified that were not initially foreseen, such as the integration with Zotero for reference management, which were evaluated and progressively incorporated into the backlog. This flexibility responds to SCRUM's own principle of adaptability, enabling a functional evolution of the system according to emerging requirements.

3.3.2. Sprint development

The technical development was carried out in an iterative and incremental manner, assigning to each sprint specific goals aimed at the progressive fulfillment of key functionalities:

1) Sprint 1. System analysis and modeling: Functional requirements were established from user interviews and represented by UML diagrams using PlantText. The general architecture of the system was defined, consisting of a frontend in Angular, a backend in Spring Boot, and a MySQL database. In addition, persistence was modeled

using JPA and an entity-relationship diagram was generated in MySQL Workbench.
2) Sprint 2. Technical environment configuration: The necessary tools for development were configured, including Java JDK, Node.js, Angular CLI, Spring Boot, Maven, MiKTeX, and MySQL. The base structure of the project was created and the connection between frontend and backend was established using REST services, verified by testing in Postman.
3) Sprint 3. Frontend development: The user interfaces corresponding to registration, login, navigation, project management, and document upload were implemented. HTTP services were developed for interaction with the backend, prioritizing the usability, visual clarity, and modularity of the components.
4) Sprint 4. Backend development: Business logic was developed, along with REST controllers, services, and repositories using Spring Data JPA. A JWT-based authentication and authorization system was implemented, and persistent management of users, projects, sections, and documents was completed. The functionalities were validated through manual testing.
5) Sprint 5. Advanced functionalities: Integrated AI services using the OpenAI API to assist in writing, analyzing, and extracting content from PDF files, generating flowcharts, and exporting documents to PDF and Word using automated LaTeX compilation. This intelligent assistance is implemented through pre-trained language models that offer contextual feedback and automatic academic text generation, simulating an intelligent writing assistant aligned with the IMRaD structure.
6) Sprint 6. Final integration and adjustments: Built-in Zotero integration for bibliographic management, developed an advanced LaTeX editor with real-time preview, and performed extensive functional testing, bug fixes, visual enhancements, and final technical adjustments.

3.3.3. Acceptance, quality, and performance testing

To ensure compliance with functional requirements, acceptance tests based on user stories were designed. These acceptance tests are shown in Table 5. Each story considered a situation representative of the use of the system, with clearly defined success criteria. These tests simulated real user interaction and validated aspects such as correct data management, response times, and expected error behavior.

Additionally, performance tests were carried out using Apache JMeter (v. 5.6.3) to evaluate the behavior of the system under high load conditions. A scenario of 1000 concurrent virtual users was simulated, with a ramp-up time of 10 s. The tests included critical actions such as simultaneous login, parallel project creation, concurrent uploading of PDF files, and bulk generation of documents in LaTeX.

Key metrics collected during these tests included:

1) Response time: Average between the sending of the request and the complete receipt.
2) Latency: The interval from the sending of the request to the first byte received.
3) Success rate: Proportion of applications successfully processed without errors.

The results showed stable performance, with acceptable response times and no loss of service, in accordance with the quality guidelines defined by international standards such as ISO/IEC 25010, ISO/IEC 25023, ITU-T G.1010, and W3C recommendations.

Table 5
User story matrix

User story	Description	Acceptance criteria
Login	Registered user logs in with credentials	Access with valid data; Incorrect credentials failed
User registration	New user creates account	Data validation; rejection of duplicates; Password confirmation
Project creation	User creates and manages projects	Full editing and viewing with mandatory data
AI support	User receives automatic help in writing	Relevant suggestions; Possibility of acceptance or rejection
Resource load	User uploads PDF files	Successful loading; AI consultation on content
Document generation	Export articles to LaTeX and PDF	Error-free compilation; file generation; Status messages
Logging out	User logs out to protect information	Login redirection; Rear access lock without authentication

3.3.4. Textual quality analysis between pre-release and proofread versions

In order to evaluate the real impact of the tool on the quality of the academic texts produced, a comparative analysis was carried out between a draft generated by a user with a basic level using RedactFlow and its expert-corrected final version.

The procedure contemplated the following stages:

1) Drafting: The user used system functionalities such as templates, recommendations, workflows, and bibliographic assistance to generate an initial academic paper.
2) Expert review: Three specialized evaluators reviewed the draft and made corrections in style, coherence, grammar, and technical adequacy.
3) Anonymous evaluation: Both versions were anonymized to avoid bias in the evaluation.
4) External grading: Three independent reviewers graded the texts using a rubric based on five criteria: consistency, clarity, structure, grammatical correctness, and academic use, with scores from 1 (low) to 5 (high).
5) Comparative analysis: The results were averaged and compared, showing significant improvements attributable to the use of the tool.

This analysis confirmed that the system provides tangible value to the initial processes of scientific writing, offering the user a solid foundation that, complemented by expert review, leads to better-structured, clearer texts that are more appropriate to academic language.

3.4. Usability evaluation

The evaluation of the usability of the web tool was designed as a process structured in two phases: a preliminary pilot test and a formal evaluation with users, with the aim of identifying interaction problems, validating the functional effectiveness of the interfaces and collecting qualitative and quantitative information on the user experience.

3.4.1. Preliminary evaluation (pilot test)

The initial phase consisted of a pilot test conducted with five selected users, following the recommendation of Virzi [21], who suggests that this number is sufficient to identify at least 80% of critical usability issues. Participants performed representative tasks within the system, while their interactions were observed and their qualitative impressions were collected.

Each session was recorded with prior informed consent and subsequently analyzed to identify barriers to use, confusion in navigation, or unintuitive functions. This information was essential to make preliminary adjustments before moving on to the formal evaluation.

3.4.2. Formal usability evaluation

In the second stage, a more extensive test was conducted with at least ten users, following the methodological guide proposed by Fox [22] to obtain statistically representative results. The participants, selected for convenience and affinity with the end-user profile (novice researchers in academic writing), completed a set of specific tasks involving the main functionalities of the system. To ensure the quality and depth of the assessment, the following instruments were used:

1) Recording consent form: Document signed by users that authorized video and screen capture during sessions.
2) Formal invitation with instructions: Document with clear guidelines for the entry and development of the assigned tasks.
3) Task list: Detailed set of actions to be executed, designed to cover the critical functionalities of the system.
4) Post-test information form (IPT): Questionnaire that collected qualitative insights at the end of the test, including open-ended and closed-ended questions.
5) System Usability Scale (SUS): Standardized instrument for measuring usability, composed of 10 items with a five-point Likert scale.

Each session was assisted by an observer who recorded the relevant behaviors and resolved minor technical doubts, without intervening in the development of the tasks. The methodological design ensured the replicability of the tests and the systematic collection of valid data for subsequent analysis.

A total of 10 novice researchers participated in the usability evaluation, obtaining an average SUS score of 77.3, classified as "good" according to international usability standards. The results also showed a reduction in average writing time from 8.2 to 4.7 h and an improvement in text quality from 2.74 to 4.15 on a five-point scale.

This study reports consolidated empirical evaluation results, including the SUS questionnaire (n = 15; mean = 76.75), 13 acceptance tests in six functionalities (100% success rate), and performance tests with 1000 virtual users (response time of 48 ms, latency of 16 ms, 100 % success rate). These results demonstrate measurable improvements in the usability, stability, and performance of RedactFlow among novice users.

4. Results

This section presents the results obtained from the application of the methodology described. The bibliographic analysis carried out to support the techniques implemented is presented, as well as the results of the development of the system and the tests carried out to validate its operation. In addition, usability and performance evaluations are included that made possible the measurement of the effectiveness and satisfaction of the user with the developed tool.

4.1. Systematic Mapping Study (SMS)

In this phase, an exhaustive bibliographic review was carried out to identify scientific articles that presented methodologies, techniques, and evidence-based recommendations for the improvement and writing of scientific texts. The objective was to ensure that the suggestions generated by the AI tool were based on reliable and recognized sources within the academic community. The search was applied in four specialized databases: IEEE Xplore, ACM Digital Library, ScienceDirect, and SpringerLink, where a total of 106 articles related to the topics of interest were located. Through a pre-selection process based on the review of abstracts, 10 studies were identified as potentially relevant for their alignment with the key terms of the project. Subsequently, a more rigorous selection was made to determine those primary articles that would contribute the greatest value to the development of the system, resulting in a total of 9 selected studies. Table 6 summarizes the number of articles found, shortlisted, and finally selected as primary studies in each database.

Table 6
Matrix of found items

Database	Found	Preselected	Primary education
IEEE Xplore	2	0	0
ACM Digital Library	2	1	1
ScienceDirect	11	3	3
SpringerLink	91	6	5
	106	10	9

Once the primary studies were defined, the techniques and methodologies addressed in each one were extracted, classifying them according to their applicability to the functional development of the system. The coding of these techniques made it possible to establish a conceptual basis for the functionalities integrated in the tool. Table 7 presents a categorized summary of the main techniques detected.

The detailed analysis of the selected literature provided a solid basis for integrating best practices in scientific writing within the developed web system. This integration ensures the generation of coherent, accurate texts aligned with academic best practices, while facilitating an effective and satisfactory user experience. Thus, the tool meets the most demanding academic standards, contributing significantly to the efficiency and quality in the preparation of scientific articles.

4.2. User interface design

As a result of the application of the user-centered design (UCD) approach, in accordance with ISO 9241-210:2019, the interfaces of the web tool were developed in three structured phases: understanding the context of use, specifying requirements and generating design solutions. This methodology made it possible to obtain a set of functionalities aimed at satisfying the real needs of novice researchers in academic writing, coherently integrated in a clear, adaptable and interactive interface.

Table 7
Techniques identified in the selected primary studies

Technique	Description
Interactive metadiscourse	Use of transitions, frame markers, glosses, evidentials, etc.
Interactional metadiscourse	Inclusion of attenuators, emphasis, attitude and commitment markers.
Rhetorical strategies	Discursive techniques for the logical organization of textual content.
Hedges and boosters	Categorization of attenuators and intensifiers into argumentative sections.
Diagnostic evaluation of discursive competence	Instruments to analyze composition and performance in scientific writing.
Global and local coherence assessment	Structural analysis of the text at macro and micro levels.

Table 8
Problems detected in academic writing and functional solutions integrated into the system

Problem identified	Proposal to fix the problem (software functionality)
Difficulty structuring articles	AI workflow assistant to guide writing by sections
Need for specific templates	LaTeX template management and customization module
Advanced use of LaTeX	Built-in automatic LaTeX editor and compiler
Finding support articles	Integration with bibliographic reference search engines
Lack of coherence in ideas	AI content organization suggestions
Difficulty in simplifying content	Assisted writing for synthesis and textual clarity
Incorrect use of terminology	Contextual academic terminology suggestions
Need for paraphrasing	AI-assisted rewriting feature
Textual fluency issues	Assistance in improving paragraph structure and logic
High learning curve in similar tools	Intuitive interface with simplified navigation
Lack of feedback	Real-time recommendations for improved consistency and style
Multiple output formats required	Export in PDF, Word and LaTeX formats
Need for automatic review	Textual analysis functionality with immediate feedback
General writing recommendations	Contextualized assistance on style and academic writing

4.2.1. Understanding and specifying the context of use

The interviews [23] with five users revealed a few common difficulties in the scientific writing process. Based on the analysis of their responses, specific needs were identified that were translated into functionalities of the system. Table 8 presents a synthesis of the main problems mentioned by users, along with the functional solutions implemented in the tool.

These findings show the importance of the interface and functionalities of responding to real problems, facilitating an efficient workflow adapted to the needs of novice researchers.

4.2.2. Specifying user requirements

From the above findings, a set of functional and non-functional requirements was defined as guided the implementation of the system. The functional requirements were organized into six large modules:

1) User management: Registration, authentication, and recovery of credentials.
2) Project management: Creating, editing, and viewing writing projects.
3) LaTeX template management: Upload, edit, and preview custom templates.
4) AI-assisted writing: Contextual assistance, text generation, paraphrasing, and auto-suggestions.
5) Resource loading and querying: Interacting with PDFs using semantic processing.
6) Document generation: Conversion and export to LaTeX, PDF and Word.
7) Regarding the non-functional requirements, the following were specified:
8) Security: JWT-based authentication, password encryption, and communication using HTTPS.
9) Usability: Responsive interface, adaptable to different devices, with minimal loading times and accessibility for users without technical experience.

4.2.3. Generation of design solutions

Based on the defined requirements, interactive prototypes were developed in Figma, which allowed them to visually represent the functionalities and navigation flows of the system. The final prototypes included the following components:

1) Authentication interface: Registration and login screens with credential validation.
2) Resource Upload Module: Panel for uploading PDF files and managing them from the main interface.
3) LaTeX template management: Visual editor with real-time preview of the generated document.
4) Project dashboard: Environment for creating, editing, and managing science writing projects.
5) Project workspace: Main section that brings together all the writing and assistance tools.
6) AI module: Built-in text analysis, idea generation, paraphrasing, and reference search capabilities.
7) LaTeX Advanced Editor: Specialized tool for experienced users, with support for syntax and validation.

These solutions were validated through internal iterations and refined based on feedback collected during usability testing. The result was a functionally complete interface, visually clear and adapted to the workflow of the target users.

4.3. Feature development

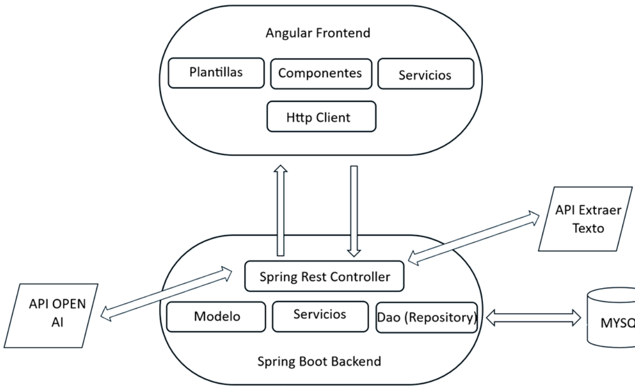
The development of the RedactFlow system was carried out in an iterative and incremental way, distributed in six sprints that allowed the progressive construction of the different functional modules. This methodology facilitated a staggered implementation, starting with the technical configuration of the environment and moving towards more complex components, such as the integration of AI, task automation, and academic writing support tools.

4.3.1. System architecture

RedactFlow was designed under a client-server architecture. The frontend was implemented using Angular, and is composed of modular components, services, and templates that communicate with the backend via HTTP RESTful requests. The backend, developed in Spring Boot, integrates business logic, REST controllers, services, and repositories connected to a MySQL relational database. The architecture also includes integration with external services, such as OpenAI's API, used for writing assistance, and an additional API for automatic extraction of textual content from PDF files. Figure 1 illustrates the logical architecture of RedactFlow, showing how the Angular frontend interacts with the Spring Boot backend through REST services, which coordinate business modules, the MySQL repository, and external APIs to support the complete manuscript preparation workflow.

The selection of the Angular and Spring Boot frameworks was based on their recognized stability, scalability, and compatibility with RESTful architectures. These technologies enable fluid communication between system layers and offer native support for microservices, facilitating maintenance and functionality expansion. In addition, their

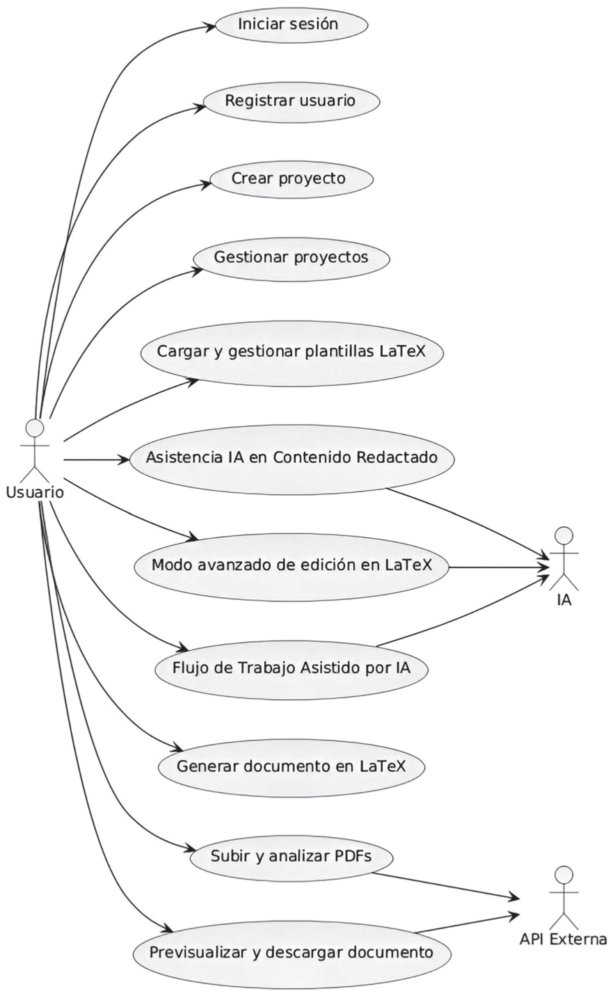
Figure 1
Overall RedactFlow system architecture



modular structure facilitates integration with intelligent services and external APIs, such as OpenAI, Zotero, and LaTeX, enabling process automation and the incorporation of intelligent assistance within the RedactFlow environment.

During the development of RedactFlow, challenges were encountered in integrating the GPT model and synchronizing microservices, which were resolved by applying incremental refactoring and iterative validation cycles.

Figure 2
Top use cases of the RedactFlow system



4.3.2. Use case modeling

The functional requirements defined during the previous phases were represented by use case diagrams using UML notation (see Figure 2). These diagrams were developed with the PlantText tool, based on PlantUML, and served as the basis for planning and implementing key system interactions.

4.3.3. Final results of software development

The RedactFlow system has been designed to offer intuitive and efficient user experience, covering the entire workflow, from initial registration to advanced academic project management. Each module was developed with the aim of ensuring a smooth and coherent interaction. The interface accessibly integrates functionalities that simplify user tasks, by providing tools that automate key processes and improve the quality of written production.

The main functionalities implemented are described below: secure access through token authentication; structured project management with the possibility of editing and reusing content; automatic upload and processing of PDF files; AI-based academic writing assistance, including generation, paraphrasing, and suggestions; a built-in LaTeX editor with real-time preview; integration with bibliographic reference managers such as Zotero; and automatic conversion of Word documents to standardized scientific templates (IEEE, APA, Springer, ACM, among others).

From this set of features, two representative screenshots are presented below as examples, illustrating the registration process (Figure 3) and the intelligent assistance functionality (Figure 4).

4.3.4. Acceptance, quality, and performance test results

In order to evaluate the stability, functional efficiency and quality of the texts generated by RedactFlow, three types of validations were carried out: acceptance tests, performance tests and textual quality analysis.

Acceptance testing: Acceptance testing was structured based on the user stories defined during the analysis stage. 13 test cases were evaluated grouped into six key functionalities. The execution in the production environment showed 100% compliance with the functional criteria, which shows a stable implementation aligned with the expected requirements. Table 9 presents the consolidated results.

Performance results: Apache JMeter v.5.6.3 was used to evaluate the system's capacity under high concurrency conditions. A scenario was set up with 1000 virtual users and a ramp-up period of 10 s. The tests included critical interactions such as mass authentication, concurrent project creation, and concurrent document processing. The results are summarized in Table 10.

Figure 3
Registration and authentication

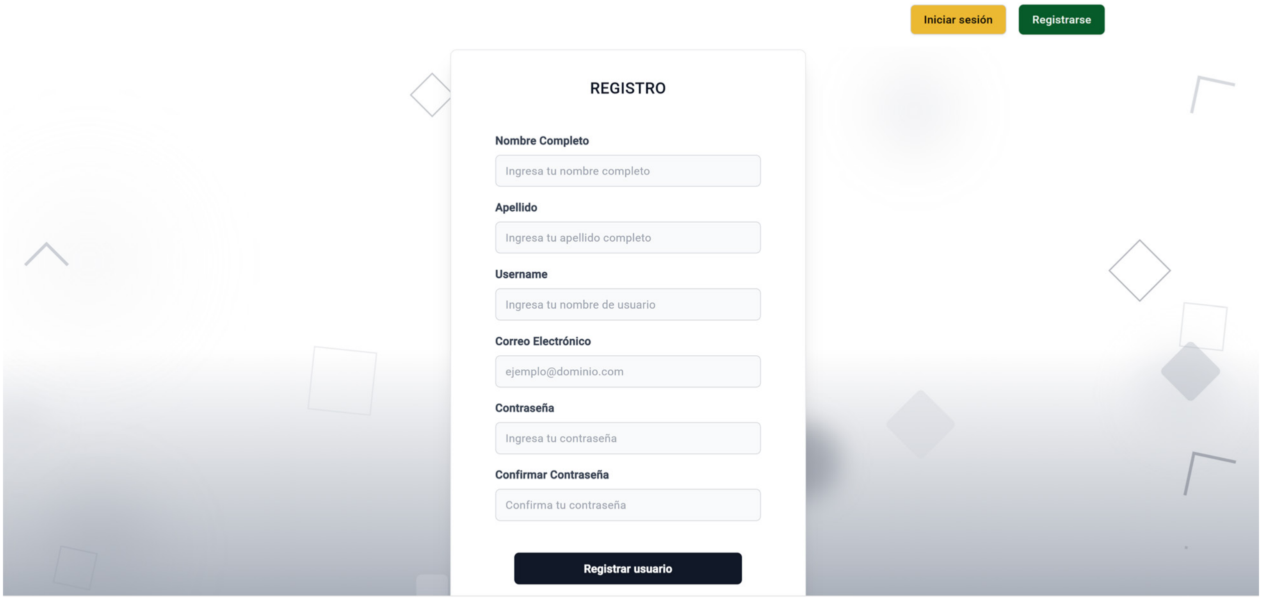
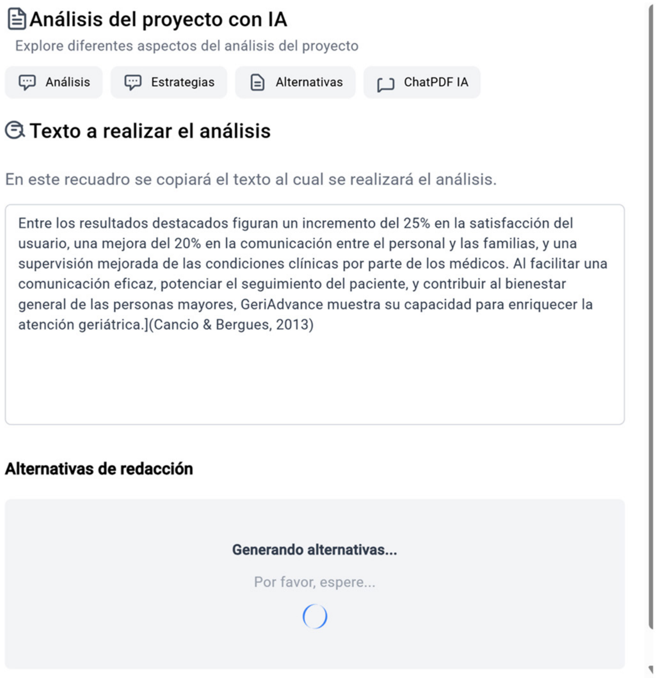


Figure 4
Artificial intelligence support



These results show that RedactFlow operates stably and efficiently even under stress scenarios, complying with the quality standards set by ISO/IEC 25010 and the performance recommendations for modern web applications.

Textual quality analysis: A comparative analysis was carried out between a preliminary version written by a user using RedactFlow and its version corrected by experts. The assessment was based on five textual quality criteria and was carried out by independent reviewers. The results are detailed in Table 11.

The results indicate that RedactFlow offers a valuable structured foundation for academic writing, especially useful for novice researchers. The most significant improvements were observed in the clarity of language and grammatical correctness, confirming their contribution in the initial phases of the writing process.

4.4. Usability evaluation

The results obtained from the application of the SUS questionnaire show a positive perception on the part of the ten users about the usability of the system. The individual scores ranged from 65 to 85 points, with an overall average of 76.75, a value that is within the range of satisfactory usability. Notably, 80% of participants exceeded the 70-point threshold, indicating an acceptable or excellent level based on the standards set out on the SUS.

Additionally, a qualitative collection of observations and suggestions was made by the participants at the end of the test. These observations, obtained from direct interaction with the tool, made it possible to identify specific aspects of improvement related to the interface and user experience. Table 12 summarizes the main problems detected, their level of priority and the recommendations proposed by users

These findings indicate that although RedactFlow achieved an overall positive level of usability, there are clear opportunities to make targeted improvements in terms of visual organization, feature redundancy, and accessibility of key elements. The suggestions collected are a valuable source of information to guide future interventions of the design and optimize the user experience.

In the tests conducted in two phases, a pilot with 5 users and a formal evaluation with 10 novice researchers in scientific writing, the tool reduced the average drafting time of an article draft from 8.2 h to 4.7 h, while increasing the textual quality score from 2.74 to 4.15 on a 1-to-5 scale. Usability, measured with the SUS questionnaire, reached an average score of 77.3, which is considered "good" according to international usability standards.

The comparative results are presented in Table 13, which contrasts the functionalities, limitations, and advantages of RedactFlow against widely used tools such as Grammarly, ChatGPT, and Paperpal.

The main indicators of efficiency, usability, and performance are summarized in Tables 9–12, demonstrating the quantitative validity of the results. These quantitative results confirm the effectiveness of the system in reducing writing time and improving text quality.

Table 9
Acceptance test results

Functionality evaluated	Total cases evaluated	Successful cases	Success rate	Highlights
Login and authentication	2	2	100%	Invalid credentials were successfully detected and secure access was allowed.
New user registration	3	3	100%	The system reacted correctly to weak passwords and duplicate users.
Logging out	2	2	100%	The login was redirected and expired sessions were not allowed.
Artificial intelligence support	2	2	100%	The suggestions and paraphrases generated were relevant and contextualized.
PDF upload and processing	2	2	100%	The files were successfully uploaded and the AI responded to queries about their contents.
Document generation	2	2	100%	The LaTeX and PDF files were generated correctly and properly notified in case of error.

Table 10
Performance test results

Metric	Average value	Interpretation
Response time	48 ms	Fluidity suitable for real-time interactive use.
Latency	16 ms	Low congestion in first response, efficient processing.
Success rate	100%	No failed request, high system reliability.

Table 11
Textual quality analysis result

Criterion	Draft	Corrected version
Content consistency	3.0	4.3
Clarity of language	2.4	4.1
Document structure	3.2	4.2
Grammar correction	2.1	4.0
Use of academic language	3.0	4.2
Overall average	2.74	4.16

Table 12
Reported usability issues and suggested improvements

Aspect	Priority	User suggestion
The "compile" and "generate draft" buttons perform the same action	Stocking	Merge the two into a single contextualized function
Unable to directly insert paraphrased text	Stocking	Incorporate option to replace the original text automatically
The references section is at the bottom of the interface	Loud	Relocate it to a quick-access side panel

Unlike these tools, which mainly focus on isolated aspects such as grammar correction, general text generation, or academic style suggestions, the proposed system offers a fully integrated academic workflow that combines structured content generation, contextualized feedback, and automatic reference management with LaTeX compatibility in a single environment. This integration directly addresses critical gaps identified in the literature, such as the absence of section-based structuring, pedagogical guidance, and compliance with editorial standards, achieving measurable improvements in efficiency, textual quality, and usability.

Table 13
Comparison of RedactFlow with existing tools

Tool	Key functionalities	Limitations	Advantages of the proposed system
Grammarly	Grammar and style correction	No article structuring, no reference management	Complete workflow with section-based structuring and automatic references
ChatGPT	General text generation	Lack of pedagogical guidance and editorial standardization	Pedagogical guidance and compliance with editorial standards
Paperpal	Academic style suggestions	No reference management or structured generation	Full integration in a single working environment

5. Discussion

RedactFlow integrates AI techniques, specifically NLP and pre-trained generative models, using the OpenAI API. These technologies support the user in key tasks in academic writing, such as structuring sections, automated grammar correction, contextualized paraphrasing, and generating consistent content. Unlike tools such as ChatGPT or Grammarly, which focus on superficial proofreading of text, RedactFlow offers a guided, structured experience adapted to the academic editorial flow, facilitating not only writing, but also learning for the user during the process.

Unlike widely used tools such as Grammarly, ChatGPT, or Paperpal, which focus mainly on isolated aspects such as grammar correction, general text generation, or academic style suggestions, RedactFlow provides a fully integrated academic workflow that unifies structured content generation, contextualized feedback, and automatic reference management with LaTeX compatibility in a single environment. This integration directly addresses critical gaps identified in the literature, including the absence of section-based structuring, pedagogical guidance, and compliance with editorial standards.

The novelty of the system lies in unifying these capabilities within an accessible platform designed specifically for novice researchers, leading to measurable improvements in textual quality (an increase in the average score from 2.74 to 4.16) and achieving a high usability rating (76.75 SUS points). This demonstrates a tangible impact on both the learning process and the final quality of scientific manuscripts.

Compared to previous studies, where limitations are identified in terms of feedback transparency and the absence of explicit pedagogical functions, RedactFlow proposes a comprehensive solution with assisted feedback and structured generation of scientific documents, including compatibility with LaTeX and bibliographic management via Zotero. These characteristics strengthen its applicability in educational settings where autonomous academic writing is sought to be encouraged, especially among novice researchers [4, 5].

The functional results show 100% compliance in the acceptance tests of the key functionalities, which reflects the technical soundness of the system and its alignment with the established requirements. Likewise, the textual quality analysis between preliminary and corrected versions showed substantial improvements in criteria such as coherence, clarity, and grammar, increasing the average score from 2.74 to 4.16. These findings confirm that RedactFlow brings tangible value in the early stages of the writing process, by providing a structured foundation on which the user can develop higher-quality texts. Compared to the traditional manual workflow for preparing articles, which requires switching between multiple applications for formatting, citation, and review, users reported a noticeable reduction in overall writing effort, greater consistency in the final document, and higher satisfaction with the automation of repetitive tasks and the feedback generated by RedactFlow.

In terms of performance, the RedactFlow platform demonstrated high stability under load conditions with 1000 concurrent users, with response times and latency within the optimal ranges set by standards such as ISO/IEC 25010. This behavior suggests that RedactFlow is scalable and suitable for institutional use in high-demand contexts.

From an interaction perspective, the results of the SUS questionnaire showed a positive perception from the users, with an average score of 76.75. This value places RedactFlow within the range of satisfactory usability, which is consistent with the principles of user-centered design (ISO 9241-210:2019) that guided its development. However, the post-test evaluation revealed opportunities for improvement in aspects such as feature redundancy, the location of critical elements (such as the reference section), and the need for more direct interaction with the paraphrased content. These observations will be incorporated into future iterations of the design, under a continuous improvement approach.

Beyond its technical capabilities, RedactFlow has significant educational potential. By offering contextualized linguistic suggestions, structural orientation, and compatibility with publication standards, it is configured as a support tool for research training. This approach is aligned with pedagogical frameworks such as self-regulated learning and formative feedback, as it promotes reflection on the text, the recognition of common errors, and the progressive improvement of writing performance. In this sense, RedactFlow not only acts as a writing assistant, but as a facilitator of academic development. The versatility of RedactFlow enables its use in the fields of Social Sciences, Education, Engineering, and Health, promoting interdisciplinary research training through intelligent assistance and adaptable templates.

Finally, although RedactFlow has demonstrated effectiveness in multiple dimensions, challenges remain related to the improvement of the argumentative style and the precise adaptation to the specific disciplinary vocabulary, aspects that require expert human intervention. Thus, the system is conceived as a valuable complement to scientific writing, but not as a replacement for academic supervision. Its greatest strength lies in the structured support it offers to those who are new to scientific production, especially in contexts where access to specialized mentors or tutors is limited.

Unlike collaborative editors such as Overleaf, Authorea, or SciFlow, RedactFlow integrates institutional and regional journal templates (e.g., SciELO and Redalyc), multilingual support (including Portuguese), and empirical usability validation (SUS = 76.75), while maintaining full compatibility with LaTeX, Word, and Zotero. Unlike platforms focused exclusively on collaborative editing, RedactFlow provides guided, section-based pedagogical workflows and standardized export options that help reduce writing time and enhance textual quality. This comprehensive design highlights its potential as both a pedagogical and technical tool for researchers in the early stages of scientific publishing.

Among the main limitations identified are its dependence on Internet connectivity, the absence of offline functionality, and the limited multilingual corpus. These areas are expected to be improved in future updates. Furthermore, the source code and a demo version are currently being prepared for publication in the institutional repository of Quevedo State University (UTEQ), ensuring transparency, reproducibility, and validation by the scientific community [24].

6. Conclusion

RedactFlow is consolidated as a comprehensive tool to assist novice researchers in the initial stages of scientific writing, combining NLP techniques, user-centered design, and adaptable modular architecture. The tool is structured around three fundamental pillars: (1) structured assistance to academic writing through guided workflows and LaTeX templates, (2) automated feedback based on AI to improve the clarity, coherence, and grammatical correctness of texts, and (3) an intuitive interface that facilitates use even among users with limited technical experience.

The results show that RedactFlow outperforms previous solutions such as Overleaf and Authorea in terms of efficiency and textual consistency by integrating linguistic analysis and pedagogical guidance into a single workflow. This work provides an open and reproducible framework for AI-assisted academic writing, applicable in both educational and institutional settings.

The development methodology used, based on adapted SCRUM, enabled an iterative advancement that incorporated constant feedback from real users and quality criteria established by international standards such as ISO 9241-210:2019 and ISO/IEC 25010. The comparative analysis between texts produced with RedactFlow and expert-corrected versions showed substantial improvements in coherence, clarity, and academic style, with an increase in the average textual quality score from 2.74 to 4.16. These results show that the tool not only automates tasks but also facilitates progressive and structured learning.

The usability evaluation, using the SUS questionnaire, yielded an average score of 76.75, indicating a positive perception of the user experience, ease of use, and general usefulness of the platform. Likewise, performance tests with up to 1000 concurrent users showed stable and efficient behavior, validating its scalability for institutional contexts.

As a future work, it is proposed to extend the pedagogical capabilities of RedactFlow by incorporating a progress tracking module that allows for the evaluation of the evolution of the user's writing skills over time. In addition, native integration with LMS platforms such as Moodle is planned to facilitate its adoption in formal educational environments. The need to incorporate advanced mechanisms of semantic personalization to adapt linguistic suggestions to the specific disciplinary vocabulary of each area of knowledge is also recognized.

Finally, future iterations will need to address challenges related to the quality of argumentative style, neutralizing biases in AI models, and improving pedagogical feedback. To this end, it is proposed to complement the current models with multilingual academic corpora and longitudinal evaluation strategies. In this way, RedactFlow will be able to continue to evolve as a solid, ethical, and effective solution for the development of scientific writing skills.

This answers the research question posed in this study, as it was demonstrated that a web workflow tool assisted by NLP techniques significantly improves the structure, clarity, and coherence of scientific articles prepared by novice researchers, facilitating a more efficient, guided, and pedagogical writing process.

Ethical Statement

This study was reviewed and approved by the Research and Ethics Committee of the Technical State University of Quevedo (Universidad Técnica Estatal de Quevedo, Ecuador) on October 30, 2024 (Certificate No. CERT-ÉTICA-024-DICYT-2024). The committee determined that the study did not involve clinical experimentation or invasive procedures with human beings, in accordance with the national regulations for Ethics Committees in Human Research (CEISH; Registro Oficial No. 279, July 1, 2014) and therefore authorized its implementation. The study involved non-invasive activities focused on software usability evaluation and educational technology. All participants were adult volunteers (novice researchers and academic users) who provided written informed consent prior to their participation. No personal, clinical, or sensitive health data were collected at any stage of the study. Data were collected anonymously and used exclusively for academic and research purposes. The study was conducted in accordance with the ethical principles of the Declaration of Helsinki. The System Usability Scale (SUS) is a publicly available instrument and does not require specific authorization for academic use. The post-test information questionnaire was designed by the authors for this study and did not require additional permission.

Conflicts of Interest

The authors declare that they have no conflicts of interest to this work.

Data Availability Statement

Data sharing is not applicable to this article as no new data were created or analyzed in this study.

Author Contribution Statement

Lucrecia Llerena: Conceptualization, Methodology, Formal analysis, Investigation, Resources, Data curation, Writing – original draft, Writing – review & editing, Visualization. **Nancy Rodríguez:** Software, Validation, Investigation, Supervision, Project administration. **Kelvin Estrada:** Methodology, Software, Validation, Formal analysis, Investigation, Resources, Data curation, Writing – original draft, Writing – review & editing, Visualization.

References

[1] Arnold, M., Goldschmitt, M., & Rigotti, T. (2023). Dealing with information overload: A comprehensive review. *Frontiers in Psychology*, *14*, 1122200. https://doi.org/10.3389/fpsyg.2023.1122200

[2] Benites, F. (2023). Information retrieval and knowledge extraction for academic writing. In O. Kruse, C. Rapp, C. M. Anson, K. Benetos, E. Cotos, A. Devitt, & A. Shibani (Eds.), *Digital writing technologies in higher education: Theory, research, and practice* (pp. 303–315). Springer. https://doi.org/10.1007/978-3-031-36033-6_19

[3] LeBlanc, A. G., Barnes, J. D., Saunders, T. J., Tremblay, M. S., & Chaput, J.-P. (2019). Scientific sinkhole: The pernicious price of formatting. *PLOS ONE*, *14*(9), e0223116. https://doi.org/10.1371/journal.pone.0223116

[4] Giglio, A. D., & Costa, M. U. P. D. (2023). The use of artificial intelligence to improve the scientific writing of non-native English speakers. *Revista da Associação Médica Brasileira*, *69*(9), e20230560. https://doi.org/10.1590/1806-9282.20230560

[5] Shaikh, S., Yayilgan, S. Y., Klimova, B., & Pikhart, M. (2023). Assessing the usability of ChatGPT for formal English language learning. *European Journal of Investigation in Health, Psychology and Education*, *13*(9), 1937–1960. https://doi.org/10.3390/ejihpe13090140

[6] Zhou, L., Wu, A. C., Hegyi, P., Wen, C., & Qin, L. (2024). ChatGPT for scientific writing—The coexistence of opportunities and challenges. *Journal of Orthopaedic Translation*, *44*, A1–A3. https://doi.org/10.1016/j.jot.2024.01.005

[7] Padrón Novales, C. I., Quesada Padrón, N., Pérez Murguía, A., González Rivero, P. L., & Martínez Hondares, L. E. (2014). Aspectos importantes de la redacción científica [Important aspects of scientific writing]. *Revista de Ciencias Médicas de Pinar del Río*, *18*(2), 362–380.

[8] Menéndez, M. R., González-Cantalapiedra, Z., & González-Polo, M. (2016). Problemas frecuentes en la redacción de artículos científicos [Review of the most frequent problems when writing a scientific article]. *EduSol*, *16*(57), 138–148.

[9] Sabaj, O. (2009). Descubriendo algunos problemas en la redacción de Artículos de Investigación Científica (AIC) de alumnos de postgrado [Discovering some problems in graduate students' composition of research articles]. *Revista Signos*, *42*(69), 107–127.

[10] Utami, S., & Karnedi, K. (2024). Enhancing students' writing paragraphs through Canva Magic AI. *Leksika: Jurnal Bahasa, Sastra dan Pengajarannya*, *18*(2), 105–116. https://doi.org/10.30595/lks.v18i2.23475

[11] Khalifa, M., & Albadawy, M. (2024). Using artificial intelligence in academic writing and research: An essential productivity tool. *Computer Methods and Programs in Biomedicine Update*, *5*, 100145. https://doi.org/10.1016/j.cmpbup.2024.100145

[12] Kacena, M. A., Plotkin, L. I., & Fehrenbacher, J. C. (2024). The use of artificial intelligence in writing scientific review articles. *Current Osteoporosis Reports*, *22*(1), 115–121. https://doi.org/10.1007/s11914-023-00852-0

[13] Singh, S. K., Kumar, S., & Mehra, P. S. (2023). Chat GPT & Google Bard AI: A review. In *2023 International Conference on IoT, Communication and Automation Technology*, 1–6. https://doi.org/10.1109/ICICAT57735.2023.10263706

[14] Khaleel, M., Nassar, Y., & El-Khozondar, H. J. (2024). Towards utilizing artificial intelligence in scientific writing. International *Journal of Electrical Engineering and Sustainability*, *2*(1), 45–50.

[15] Chetwynd, E. (2024). Ethical use of artificial intelligence for scientific writing: Current trends. *Journal of Human Lactation*, *40*(2), 211–215. https://doi.org/10.1177/08903344241235160

[16] Khabib, S. (2022). Introducing artificial intelligence (AI)-based digital writing assistants for teachers in writing scientific articles. *Teaching English as a Foreign Language Journal*, *1*(2), 114–124. https://doi.org/10.12928/tefl.v1i2.249

[17] Kusmanto, H., Sohnui, S., Sari, E. R., & Mujiani, I. (2025). The utilization of artificial intelligence in scientific writing education. *Educative: Jurnal Ilmiah Pendidikan*, *3*(1), 35–48. https://doi.org/10.70437/educative.v3i1.798

[18] Heung, Y. M. E., & Chiu, T. K. F. (2025). How ChatGPT impacts student engagement from a systematic review and meta-analysis study. *Computers and Education: Artificial Intelligence*, *8*, 100361. https://doi.org/10.1016/j.caeai.2025.100361

[19] Patel, K., Beeram, D., Ramamurthy, P., Garg, P., & Kumar, S. (2024). AI-enhanced design: Revolutionizing methodologies and workflows. *International Journal of Artificial Intelligence Research and Development*, *2*(1), 135–157.

[20] Pea, R. D. (1987). User centered system design: New perspectives on human–computer interaction. *Journal Educational Computing Research*, *3*(1), 129–134.

[21] Virzi, R. A. (1992). Refining the test phase of usability evaluation: How many subjects is enough? *Human Factors: The Journal of the Human Factors and Ergonomics Society*, *34*(4), 457–468. https://doi.org/10.1177/001872089203400407

[22] Fox, J. E. (2015). The science of usability testing. In *Proceedings of the 2015 Federal Committee on Statistical Methodology (FCSM) Research Conference*, 1–7.

[23] Estrada, K. (2025). *Pregunta de entrevista* [Semi-structured interview questionnaire]. Cuestionario de las Entrevistas. https://n9.cl/7w56i

[24] Estrada, K. (2024). *RedactFlow: Source code repository (backend and frontend)* [Data set]. GitHub. https://github.com/kestradalp/redactFlow

How to Cite: Llerena, L., Rodríguez, N., & Estrada, K. (2025). Development of a Web Workflow Tool That Facilitates and Optimizes the Process of Preparing Scientific Articles. *Artificial Intelligence and Applications*. https://doi.org/10.47852/bonviewAIA52027141