Thoth 2.0: Advancing an RSL Tool for Enhanced Snowballing Support

Diego Comis
diegocomis@gmail.com
Instituto Federal Farroupilha
Santa Maria, RS, Brazil

Elder de Macedo Rodrigues
elderrodrigues@unipampa.edu.br
Federal University of Pampa
Alegrete, Brazil

ABSTRACT

Thoth is an open-source, web-based tool designed to support researchers in conducting Systematic Literature Reviews (SLRs) by automating critical phases of the review process. Currently in its final testing and refinement stage, the tool introduces an advanced Snowballing module that systematically identifies relevant studies through backward and forward citation analysis, integrated with the CrossRef API for reliable metadata retrieval. By automating repetitive tasks, including study selection, quality assessment, and data extraction, Thoth 2.0 significantly reduces manual effort while minimizing human error. Its user-friendly interface supports collaborative workflows, enabling geographically dispersed teams to coordinate SLRs seamlessly. As a freely available solution, Thoth 2.0 addresses key limitations of existing SLR tools, offering a unified platform that combines automation, rigorous methodology, and adaptability to diverse research needs. Its Snowballing module, in particular, extends literature coverage with minimal researcher intervention, demonstrating the tool's potential to enhance the efficiency and reproducibility of evidence synthesis.

Thoth demo video: https://doi.org/10.5281/zenodo.15493167

KEYWORDS

Thoth, SLR Tool, SMS, Snowballing, Software

1 Introduction

Systematic Literature Reviews (SLRs) are a fundamental practice in evidence-based research, providing rigorous methodologies to synthesize existing knowledge of a research area. Among the techniques employed in SLRs, snowballing—a method for iteratively identifying relevant studies through backward (references) and forward (citations) chaining—has gained prominence for its ability to uncover literature that traditional database searches might miss [21]. However, the manual execution of snowballing is time-consuming and prone to errors, highlighting the need for tool support to automate and streamline the process.

Thoth, developed as a web-based tool for SLRs, addressed key challenges in protocol management, study selection, and data extraction [14]. While its first version provided foundational support for SLRs, it lacked integrated snowballing functionality, requiring researchers to rely on external tools or manual efforts. Existing tools like Buhos and SMS Builder offer partial snowballing support, but gaps remain in usability, integration with SLR workflows, and automation [3].

Authors' Contact Information: Diego Comis, diegocomis@gmail.com, Instituto Federal Farroupilha, Santa Maria, RS, Brazil; Elder de Macedo Rodrigues, elderrodrigues@unipampa.edu.br, Federal University of Pampa, Alegrete, Brazil.

This paper presents Thoth 2.0, an enhanced version of the tool that introduces a dedicated snowballing module. By integrating APIs like CrossRef[1] for reference retrieval and Algolia AI Synonyms[2] for synonym-based search expansion, Thoth 2.0 is intended to reduces manual effort and improves the reproducibility of snowballing. The development was driven by user feedback from a usability study [5], which identified snowballing as a critical unmet need. Our work aligns with broader efforts to bridge the gap between SLR tools and iterative search methods, offering a unified platform for comprehensive SLRs.

1.1 Problem statement

The Thoth tool was initially developed by undergraduate Software Engineering students to address gaps identified in existing tools, as highlighted by [14]. The primary motivation for its development was the need to integrate critical features that were inadequately supported by existing solutions. The tool's design was guided by five core principles: **(1)** open-source availability and free access, ensuring inclusivity and community contribution; **(2)** continuous maintenance to uphold reliability and relevance; **(3)** a user-friendly interface to promote accessibility and ease of use; **(4)** cross-platform compatibility for broader applicability; and **(5)** essential functionalities tailored to support systematic reviews (SLR), addressing the specific needs of the evidence-based research community.

Over time, limitations in the original version of Thoth became apparent, as identified in the usability evaluation conducted by [5]. Issues reported included a lack of documentation and help resources, unclear error messages, a complex and unintuitive interface, and missing functionalities such as an integrated Snowballing module. Additionally, technical issues such as session expirations without warning, page reload dependencies, and database import failures significantly impacted the user experience and efficiency of the systematic review process.

To address these challenges, the new version of Thoth was developed with a focus on three main pillars:

(1) **Usability Enhancements:** The interface was redesigned to improve navigation, incorporating contextual guides, tooltips, and comprehensive documentation with step-by-step tutorials. Error messages were reformulated to provide clearer and more actionable feedback to users.
(2) **Bug Fixes and Stability:** Major technical issues were addressed, including database import failures, unexpected session expirations, and the need for frequent page reloads. These fixes improved system stability, ensuring a smoother and more reliable user experience.

[1] https://api.crossref.org/swagger-ui/index.html
[2] https://www.algolia.com/doc/guides/algolia-ai/dynamic-synonym-suggestions/

(3) **New Functionalities:** A Snowballing module was integrated, allowing researchers to identify additional relevant studies automatically. Other new features include password recovery, flexible data export options, and improved study search capabilities. Furthermore, APIs such as Algolia, CrossRef, and Springer were incorporated to enhance search efficiency and systematic review execution.

These improvements make Thoth a more robust and efficient tool for conducting systematic reviews, reinforcing its role in supporting the academic and professional community. By addressing previous limitations and incorporating user feedback, the new version enhances the overall research workflow, ensuring a more intuitive and feature-rich experience.

2 Background

Systematic Literature Reviews (SLRs) are essential for synthesizing research findings within a specific field, allowing researchers to identify trends, gaps, and overlooked issues in the literature [1, 11]. However, conducting SLRs manually is a labor-intensive process that demands meticulous attention to detail, particularly in study selection, quality assessment, and data extraction [7]. Although well-defined protocols exist for SLRs, the support of dedicated tools is highly desirable and often necessary. Such tools can enhance research consistency, automate article analysis, and optimize the time spent on study selection and evaluation by [15, 19]. Over the years, various tools have been developed to assist in this process, including Parsifal, SLuRp, and Porifera, each offering functionalities such as collaborative workspaces, semi-automated data extraction, and search string optimization [2, 3, 12, 14].

Despite the availability of these tools, many exhibit significant limitations, including discontinuation, high costs, or restricted functionalities. Some tools are no longer maintained, while others operate under subscription models, limiting access for researchers with constrained resources. Moreover, few integrate all essential functionalities into a single platform, often providing complementary yet non-overlapping features. This fragmented landscape presents challenges for researchers seeking comprehensive solutions for conducting SLRs efficiently.

2.1 Related Tools

Several tools have been developed to support Systematic Literature Reviews (SLRs) and Systematic Mapping Studies (SMS), each offering unique functionalities to streamline the review process. Among these, **Parsifal** [3] stands out as a web-based tool designed specifically for Software Engineering researchers. It supports all stages of the SLR process, from planning to execution, and facilitates collaboration among distributed teams. Parsifal helps define research objectives, PICOC criteria, search strings, and inclusion/exclusion criteria, making it a comprehensive solution for SLRs in software engineering. Similarly, **SMS Builder**, an open-source Java-based tool, is tailored for SMS construction. It provides functionalities for study selection, data extraction, and result synthesis, with clear instructions for running the software using Docker containers [4]. **Buhos** [2] is another notable tool, supporting both individual and group systematic reviews. It integrates with Crossref for deduplication and reference management and is one of the few tools that support the snowballing technique. Developed using the Sinatra framework, Buhos offers a user-friendly interface and robust functionalities for managing SLR protocols, making it a reliable option for collaborative research . Another powerful tool is **DistillerSR**[4], a web-based solution widely used across various research fields. It excels in features such as dual-reviewer screening, conflict resolution, duplicate detection, and PRISMA-compliant reporting. Although it requires a subscription, its comprehensive functionalities make it a strong choice for high-quality systematic reviews. **Porifera**[3] is a collaborative web-based tool designed to support both SLRs and SMSs. It allows researchers to define project goals, inclusion/exclusion criteria, and search strings, while importing publications and identifying duplicates. Porifera facilitates the selection process through individual criterion application and disagreement resolution, integrating external services for translation and reference citation. Its features aim to optimize time, prevent errors, and improve the reliability of evaluations . Another innovative tool is **ASReview**[5], which leverages active learning techniques to automate the screening process. Developed at Utrecht University, it efficiently sorts large volumes of text, reducing manual effort in the initial stages of an SLR [9, 20].

Finally, **Rayyan** is a widely used tool that offers functionalities for evaluating studies and references. It is unique in providing a mobile application, allowing researchers to work on the go. While its free version has limited features, Rayyan supports collaboration and is valued for its user-friendly interface and efficient study screening capabilities [10, 18]. These tools collectively address various aspects of the systematic review process, offering researchers a range of options to enhance efficiency, collaboration, and accuracy in their work. The comparative analysis of these tools was conducted through a systematic mapping study, and the complete dataset and results are available in the Zenodo repository [6].

3 Thoth 2.0

Thoth 2.0[6] [7] is a web-based, open-source tool designed to support researchers in conducting **Systematic Literature Reviews (SLRs)** and **Snowballing Studies**. Addressing the challenges of manual SLR processes, it offers advanced automation, a user-friendly interface, and a fully digital snowballing module, significantly improving the efficiency and accuracy of literature reviews. Built using modern technologies such as **Laravel 10**, **Livewire 3**, and **Docker**, it ensures scalability, ease of maintenance, and cross-platform compatibility. The tool follows a structured workflow divided into four main stages: **Planning**, **Study Selection**, **Quality Assessment**, and **Data Extraction & Results**, streamlining the SLR process from initial setup to final reporting, as illustrated in Figure 1.

(1) **Planning**:
- Users can create new projects and choose between three types of studies: **SLR**, **Snowballing**, or a combination of both.

[3] https://parsif.al

[4] https://www.distillersr.com
[5] https://asreview.nl
[6] https://thoth-slr.com
[7] https://github.com/Thoth2023/thoth-remake

- The tool allows researchers to define the research domain, select databases (e.g., IEEE, Springer, Scopus), and configure search strings with the help of **API Algolia AI Synonyms** for synonym suggestions.
- Criteria for inclusion/exclusion, quality assessment questions, and data extraction queries are also set during this stage.

(2) **Study Selection**:
- For **SLR**, users apply search strings to selected databases, export results in **.bibtex** format, and import them into Thoth for further processing.
- For **Snowballing**, users import a **start set** of studies, and Thoth automatically retrieves references (backward) or citations (forward) using the **CrossRef API**.
- The tool assists in removing duplicates and filtering studies based on predefined criteria.

(3) **Quality Assessment**:
- Studies that pass the inclusion/exclusion criteria are evaluated based on quality assessment questions configured during planning.
- Thoth automatically calculates quality scores, allowing researchers to classify studies according to their methodological rigor.

(4) **Data Extraction & Results**:
- Relevant data from selected studies are extracted using predefined queries.
- Thoth generates comprehensive reports and visualizations, enabling researchers to analyze and share their findings effectively.

Figure 1 illustrates the structured workflow of Thoth, divided into key stages for conducting Systematic Literature Reviews (SLRs) and Snowballing Studies. The process begins with Project Creation and Protocol Management, where researchers define research questions, search strategies, databases, and inclusion/exclusion criteria. In the Study Selection phase, search strings are applied to selected databases (e.g., Springer, Scopus), and studies are imported in .bibtex format. For Snowballing, base studies are imported, and Thoth performs backward/forward citation analysis using the CrossRef API. The tool assists in removing duplicates and applying inclusion/exclusion criteria, leading to the identification of eligible studies. These studies undergo Quality Assessment, where predefined quality questions are used to evaluate methodological rigor. Finally, data is extracted, tabulated, and presented in comprehensive reports, enabling researchers to analyze and share their findings effectively.

3.1 Software architecture

Redesigned to overcome the limitations of its predecessor, Thoth's architecture ensures scalability, maintainability, and efficiency. The new version incorporates modern technologies and frameworks, enhancing both robustness and usability. At its core, **Laravel 10** [17] provides a solid foundation, leveraging the **MVC (Model-View-Controller)** pattern to facilitate a clean separation of concerns, which simplifies maintenance and future improvements. For the frontend, **Livewire 3** [13] enables dynamic, responsive interactions without requiring full page reloads, significantly improving the user experience.

Consistency across different environments is achieved through **Docker** [8] containers, which streamline development, testing, and deployment. This containerization approach minimizes conflicts and guarantees uniform application behavior across platforms. Database management has also been modernized with **migrations**, ensuring version control for schema changes, seamless updates, and rollback capabilities. Data storage is handled by **MySQL**, a widely used and reliable relational database system.

Automated testing and deployment processes are supported by **GitHub Actions** for **CI/CD (Continuous Integration/Continuous Deployment)**, reducing errors and accelerating the development cycle. A responsive and modern user interface is maintained through **Bootstrap**, while data import and export operations are efficiently managed using **PHPSpreadSheet** and **League/CSV**. Designed with modularity and extensibility in mind, Thoth allows for seamless integration of new features. The decoupling of backend and frontend components facilitates independent development and testing, ensuring adaptability to future technological advancements and evolving user requirements.

A visual representation of the architecture is provided in Figure 2, illustrating the key components and their interactions. This structured design not only addresses previous shortcomings but also establishes a solid foundation for continued enhancements and long-term sustainability.

3.2 Requirements

- **RQ1) The tool must incorporate an integrated Snowballing module.** To meet researcher demands, the new version must include a Snowballing module that automates the identification of additional studies through backward and forward citation analysis. This module should seamlessly integrate into the systematic review workflow, minimizing manual intervention and enhancing process efficiency.
- **RQ2) The tool must integrate the CrossRef API for reference searches.** To complement the Snowballing module, the tool must integrate the CrossRef API to automate searches for references and citations related to studies included in the review, enhancing efficiency in identifying relevant literature.
- **RQ3) The Snowballing module must support both backward and forward citation analysis.** The module must allow users to retrieve references cited by a study (backward) and identify newer studies that cite the original work (forward). These functionalities must be based on DOI metadata retrieved through external APIs, ensuring a comprehensive exploration of the literature.
- **RQ4) The tool must support the Snowballing module as a standalone review type.** Researchers must be able to create Snowballing-only projects, uploading a BibTeX file as a start set and configuring inclusion and quality assessment criteria, enabling fully independent citation-based reviews.
- **RQ5) The tool must allow Snowballing to be used as an extension of an SLR.**After completing a traditional SLR

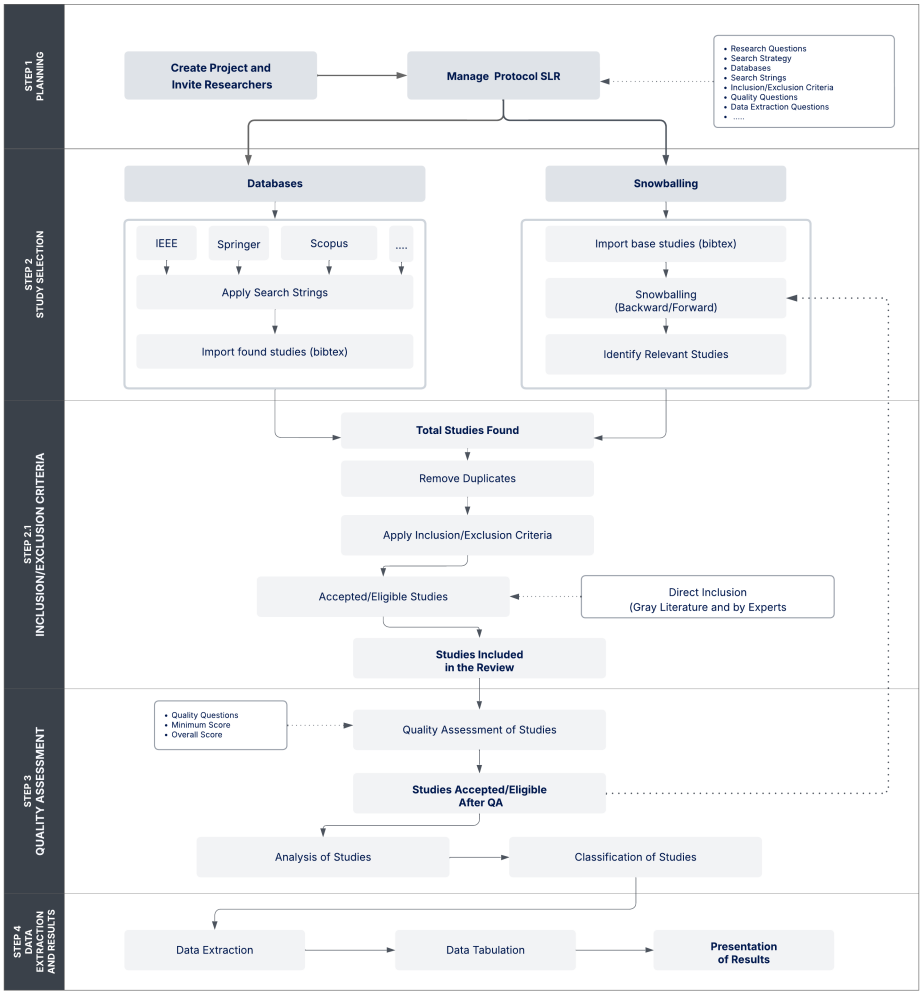

Figure 1: Workflow of Thoth 2.0 for Systematic Literature Reviews and Snowballing Studies

workflow, users must be able to apply Snowballing selectively on relevant studies to expand the scope of the review while maintaining methodological rigor.

- **RQ6) The tool must allow users to evaluate and select studies retrieved via Snowballing.** All references retrieved by the Snowballing module must be displayed in an organized interface where users can assess each entry using preconfigured inclusion/exclusion and quality criteria.
- **RQ7) The system must provide traceability and history for all references processed by Snowballing.** Every evaluated reference (included or excluded) must be logged, with decisions traceable for transparency, reproducibility, and audit purposes.
- **RQ8) The Snowballing module must perform asynchronous operations using background jobs.** Tasks such as retrieving references or updating metadata must be executed asynchronously (e.g., using Laravel Jobs) to ensure responsiveness and avoid UI blocking during processing.
- **RQ9) The tool must present real-time feedback during Snowballing tasks.** The system must provide progress indicators, success/error messages, and visual confirmation of task completion or failure, improving user experience and reliability during batch operations.

3.3 Design Decisions

This section outlines the design decisions (DDs) made during the development of the tool to address the requirements discussed earlier. Each decision is linked to its corresponding requirement.

1. Integrated Snowballing Module RQ1 — A dedicated Snowballing module was developed and embedded into the systematic review workflow, supporting both backward and forward citation analysis. **Justification:** Automates the discovery of additional studies, reducing manual work and increasing review efficiency.

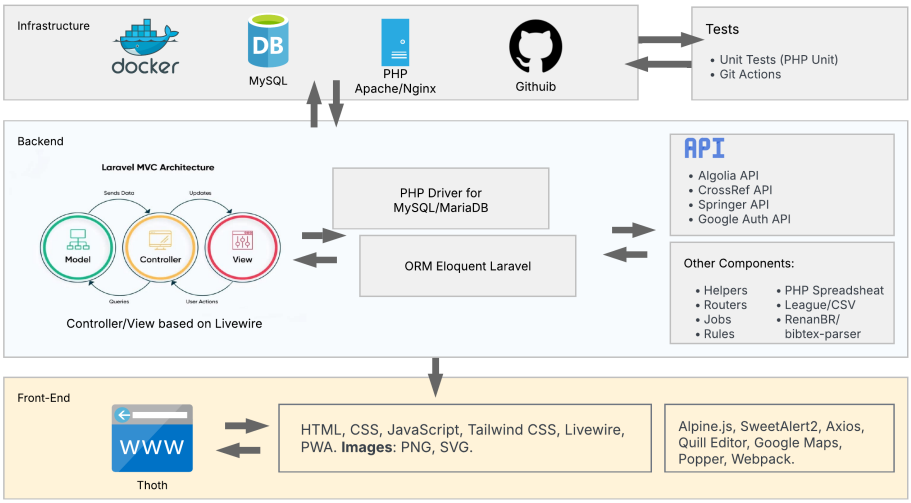

Figure 2: Architecture Thoth 2.0

2. Integration with CrossRef API for Citation Retrieval RQ2 and **RQ3** — The module integrates the CrossRef API to automatically retrieve references based on DOIs, including both backward and forward citations. **Justification:** Ensures accurate and scalable citation retrieval, improves literature coverage, and reduces manual effort during the snowballing process.

3. Standalone Snowballing Projects RQ4 — Enables creating Snowballing-only projects starting from a BibTeX *start set*. Justification: Increases flexibility for citation-based reviews.

4. Snowballing as an SLR Extension RQ5 — Allows Snowballing to be applied after primary study selection in SLRs. **Justification:** Combines SLR rigor with broader discovery.

5. Evaluation Interface for Retrieved Studies RQ6 — Provides an interface to assess studies using predefined inclusion and quality criteria. **Justification:** Ensures consistent and structured evaluation.

6. Reference History and Traceability RQ7 — Logs all reviewed citations and decisions for auditing purposes. **Justification:** Promotes transparency and reproducibility.

7. Asynchronous Task Execution RQ8 — Uses Laravel Jobs for background processing of citations and metadata. **Justification:** Improves system performance and user experience.

8. Real-Time User Feedback RQ9 — Implements status messages and indicators during citation processing. **Justification:** Enhances usability and error awareness.

3.4 The Snowballing tool

The **Snowballing** module in *Thoth 2.0* [8] [9] was designed to optimize the identification of additional studies during systematic literature reviews (SLRs) by providing an automated and integrated workflow. This module directly addresses limitations observed in the previous version, particularly the lack of citation-based discovery features, and fulfills the design requirements RQ2 and RQ3 by minimizing manual effort and enhancing efficiency.

Snowballing is a widely accepted technique in SLRs, involving the exploration of references cited by selected studies (*backward Snowballing*) and studies that cite them (*forward Snowballing*). Thoth provides two usage modes for Snowballing: (i) as a standalone review configuration and (ii) as an extension to a conventional SLR.

In the standalone mode, the user configures a Snowballing project by uploading a BibTeX file representing the *start set*. From these initial references, the user can trigger backward and/or forward searches. In backward searches, the module uses the DOI of each article to automatically retrieve its cited references using the **CrossRef API**. These references are then presented to the user for evaluation based on predefined inclusion and quality criteria, enabling methodologically consistent expansion of the study set.

Forward searches follow a similar structure, identifying newer studies that cite the originals. This approach is particularly useful for tracking recent advancements and the evolution of knowledge over time. The Snowballing module supports manual validation of all retrieved references, ensuring that only relevant and high-quality studies are incorporated.

Alternatively, Snowballing can be used after the primary study selection phase of a conventional SLR, allowing researchers to selectively expand their review based on highly relevant studies. This hybrid strategy combines the rigor of traditional SLRs with the breadth of citation-based discovery, offering greater coverage without compromising methodological quality.

To support these capabilities, the CrossRef API was integrated using secure RESTful requests. Retrieved references are parsed and presented in a responsive interface that enables filtering, inspection, and decision-making. Additionally, all user actions and study decisions are logged for transparency and reproducibility.

From a technical perspective, the system leverages asynchronous processing through **Laravel Jobs** to manage long-running tasks. The `ProcessReferences` job is responsible for fetching and

[8] https://thoth-slr.com
[9] https://github.com/Thoth2023/thoth-remake

storing reference data based on DOI, checking for duplicates, and updating metadata when available. The `AtualizarDadosSpringer` job complements this by querying the **Springer API** for additional metadata such as abstracts, keywords, and author information.

Both jobs are integrated with Livewire components, which dynamically update the user interface and provide immediate feedback. For instance, when references are fetched, the `PaperModal` controller triggers the job, processes the response, and dispatches events to update the corresponding UI components (*ReferencesTable*). This ensures that users can evaluate newly found studies without delays or the need for page reloads.

Finally, the workflow is illustrated in Figure 3, which depicts the stages involved in executing Snowballing within Thoth. Additionally, a demonstration video is provided to help users understand the full process—from configuration to the inclusion of evaluated references—highlighting the importance of this functionality in conducting rigorous and comprehensive systematic reviews.

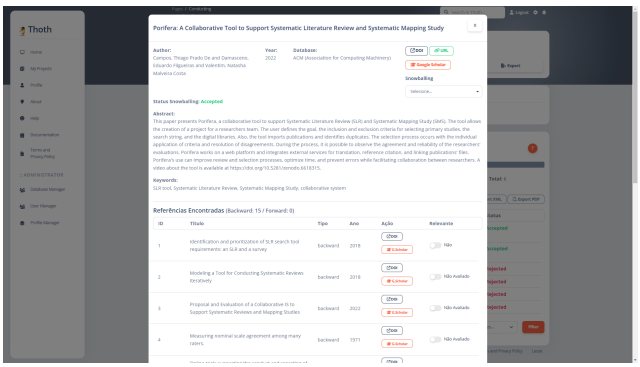

Figure 3: Snowballing Module Workflow in Thoth

4 Lessons learned and future works

The development of *Thoth 2.0* has yielded valuable insights, particularly regarding the integration of the *Snowballing* module—an innovative feature that automates backward and forward citation analysis within a collaborative SLR workflow. Currently, the module is undergoing final testing by undergraduate Software Engineering students at UNIPAMPA, with preliminary results confirming its potential to streamline study identification while reducing manual effort. Notably, *Thoth* stands out among existing tools by unifying *Snowballing* with full SLR support in a single platform, addressing a critical gap in research methodology tools.

Despite these advancements, further refinements are required to achieve a stable release. Pending tasks include resolving minor reported bugs and conducting a comprehensive evaluation of the tool's usability and performance in real-world SLR scenarios. Once completed, these steps will ensure *Thoth 2.0* meets the robustness expected by the scientific community.

As an open-source project, *Thoth* actively invites collaboration from researchers and developers to contribute innovative ideas, particularly in enhancing automation, expanding API integrations (e.g., Scopus, IEEE Xplore), and refining conflict-resolution mechanisms for multi-reviewer workflows. The project's GitHub repository and documentation are openly accessible, fostering community-driven evolution.

Future work will focus on **Empirical validation** of the *Snowballing* module's impact on SLR efficiency and coverage; **Scalability improvements** to handle large-scale reviews; **Integration of AI-assisted screening** to further reduce manual workload; **Adoption incentives**, including workshops and partnerships with research groups.

5 Significance

Thoth has significantly advanced systematic literature reviews (SLRs) by addressing key challenges and introducing innovative features that enhance research efficiency, accuracy, and collaboration. A major contribution is the automation of labor-intensive tasks such as study selection and quality assessment, previously time-consuming and prone to human error. The addition of the Snowballing module, a highly requested feature, expands research possibilities by automating the identification of additional studies through reference and citation analysis. This functionality is particularly valuable for large-scale SLRs, where manually analyzing references would be impractical. Its applicability has already been demonstrated in studies such as those by [3], [16], and [9].

Beyond research advancements, Thoth has transformed the daily practices of its users. Automating key SLR processes significantly improves efficiency, allowing researchers to focus on data analysis and interpretation rather than repetitive manual tasks. Standardized workflows enhance accuracy by minimizing human error and ensuring consistency in study selection and quality assessment. Additionally, integrated collaborative features enable geographically distributed research teams to work seamlessly.

Widespread adoption further underscores Thoth's impact. Actively used in numerous research projects and cited in several academic publications, its relevance within the research community is well established. Comparative data on studies that employed Thoth can be found in the Zenodo repository [6]. While primarily adopted by researchers in software engineering and related fields, its potential extends beyond academia. Industries such as healthcare, technology, and market research increasingly rely on systematic reviews for decision-making.

Despite its advantages, certain limitations remain. User feedback, particularly from Charles [5], has highlighted areas for improvement, including the need for better documentation, clearer error messages, and a more intuitive user interface. These insights have guided ongoing development efforts, leading to refinements that improve usability and address structural challenges. The Snowballing module, for instance, was developed in direct response to demands for greater automation and efficiency in the SLR process. Future updates will continue enhancing functionality and expanding applicability to new research domains.

ARTIFACT AVAILABILITY

The authors declare that the research artifacts supporting the findings of this study are accessible at https://doi.org/10.5281/zenodo.17064560.

REFERENCES

[1] David Budgen and Pearl Brereton. 2006. Performing Systematic Literature Reviews in Software Engineering. *SIGSOFT - Special Interest Group on Software Engineering* (2006), 1051–1052. doi:10.1145/1134285.1134500

[2] Claudio Bustos Navarrete, María Gabriela Morales Malverde, Pedro Salcedo Lagos, and Alejandro Díaz Mujica. 2018. Buhos: A web-based systematic literature review management software. *SoftwareX* 7 (2018), 360–372. doi:10.1016/j.softx.2018.10.004

[3] Thiago Prado De Campos, Eduardo Filgueiras Damasceno, and Natasha Malveira Costa Valentim. 2022. Porifera: A Collaborative Tool to Support Systematic Literature Review and Systematic Mapping Study. In *Proceedings of the XXXVI Brazilian Symposium on Software Engineering* (<conf-loc>, <city>Virtual Event</city>, <country>Brazil</country>, </conf-loc>) *(SBES '22)*. Association for Computing Machinery, New York, NY, USA, 452–457. doi:10.1145/3555228.3555273

[4] Christian A. Candela-Uribe, Luis E. Sepúlveda-Rodríguez, Julio C. Chavarro-Porras, John A. Sanabria-Ordoñez, José Luis Garrido, Carlos Rodríguez-Domínguez, and Gabriel Guerrero-Contreras. 2022. SMS-Builder: An adaptive software tool for building systematic mapping studies. *SoftwareX* 17 (2022), 100935. doi:10.1016/j.softx.2021.100935

[5] Fabienne Charles. 2022. *A study on the perception to usability of the thoth tool.* Master's thesis. Universidade Federal do Pampa, Alegrete, RS, Brazil. 137p. Dissertação (Mestrado Profissional em Engenharia de Software).

[6] Diego dos Santos Comis. 2025. *Mapeamento Sistemático da Literatura.* doi:10.5281/zenodo.14865918

[7] Luyi Feng, Yin Kia Chiam, and Sin Kuang Lo. 2017. Text-Mining Techniques and Tools for Systematic Literature Reviews: A Systematic Literature Review. In *2017 24th Asia-Pacific Software Engineering Conference (APSEC)*. 41–50. doi:10.1109/APSEC.2017.10

[8] Docker Inc. 2024. Docker Documentation. https://docs.docker.com/ Acesso em: 14 dez. 2024.

[9] Burak Karakan. 2021. *Tool support for systematic literature reviews: analyzing existing solutions and the potential for automation.* B.S. thesis. University of Stuttgart. https://d-nb.info/1232727970/34

[10] Liz Kellermeyer, Ben Harnke, and Shandra Knight. 2018. Covidence and rayyan. *Journal of the Medical Library Association: JMLA* 106, 4 (2018), 580. doi:10.5195/jmla.2018.513

[11] Barbara Kitchenham, Rialette Pretorius, David Budgen, O. Pearl Brereton, Mark Turner, Mahmood Niazi, and Stephen Linkman. 2010. Systematic literature reviews in software engineering – A tertiary study. *Information and Software Technology* 52, 8 (2010), 792–805. doi:10.1016/j.infsof.2010.03.006

[12] Christian Kohl, Emma J. McIntosh, Stefan Unger, Neal R. Haddaway, Steffen Kecke, Joachim Schiemann, and Ralf Wilhelm. 2018. Online tools supporting the conduct and reporting of systematic reviews and systematic maps: A case study on CADIMA and review of existing tools. *Environmental Evidence* 7, 1 (2018). doi:10.1186/s13750-018-0115-5 Cited by: 179; All Open Access, Gold Open Access, Green Open Access.

[13] Laravel and Tailwind CSS. 2024. Livewire Documentation. https://laravel-livewire.com/docs Acesso em: 14 dez. 2024.

[14] Luciano Marchezan, Guilherme Bolfe, Elder Rodrigues, Maicon Bernardino, and Fábio Paulo Basso. 2019. Thoth: A Web-based Tool to Support Systematic Reviews. In *2019 ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM)*. 1–6. doi:10.1109/ESEM.2019.8870160

[15] Christopher Marshall, OP Brereton, and B Kitchenham. 2018. Tool features to support systematic reviews in software engineering–a cross domain study. *e-Informatica Software Engineering Journal* (2018). https://dl.acm.org/doi/10.1145/2601248.2601270

[16] Moritz Messerschmidt, Samantha Chan, Elliott Wen, and Suranga Nanayakkara. 2022. Toro: A Web-based Tool to Search, Explore, Screen, Compare and Visualize Literature. *SIGHCI 2022* (2022). https://aisel.aisnet.org/sighci2022/13/

[17] Taylor Otwell. 2024. Laravel Documentation. https://laravel.com/docs Acesso em: 14 dez. 2024.

[18] Mourad Ouzzani, Hossam Hammady, Zbys Fedorowicz, and Ahmed Elmagarmid. 2016. Rayyan—a web and mobile app for systematic reviews. *Systematic reviews* 5 (2016), 1–10. https://link.springer.com/content/pdf/10.1186/s13643-016-0384-4.pdf

[19] Kai Petersen, Sairam Vakkalanka, and Ludwik Kuzniarz. 2015. Guidelines for conducting systematic mapping studies in software engineering: An update. *Information and Software Technology* 64 (2015), 1–18. doi:10.1016/j.infsof.2015.03.007

[20] Rens Van De Schoot, Jonathan De Bruin, Raoul Schram, Parisa Zahedi, Jan De Boer, Felix Weijdema, Bianca Kramer, Martijn Huijts, Maarten Hoogerwerf, Gerbrich Ferdinands, et al. 2021. An open source machine learning framework for efficient and transparent systematic reviews. *Nature machine intelligence* 3, 2 (2021), 125–133. https://www.nature.com/articles/s42256-020-00287-7

[21] Claes Wohlin. 2014. Guidelines for snowballing in systematic literature studies and a replication in software engineering. In *Proceedings of the 18th International Conference on Evaluation and Assessment in Software Engineering* (London, England, United Kingdom) *(EASE '14)*. Association for Computing Machinery, New York, NY, USA, Article 38, 10 pages. doi:10.1145/2601248.2601268