

Technical Validation of SREMAN: A Browser-Based Systematic Review and Evidence Management Tool for Cross-Database Interoperability, Screening-Data Preservation, and Project-State Restoration

Vishnu Shivam^{1,*}

¹Research and Development, Vedanadhi, Tamil Nadu, India-636115.

*Correspondence: Email ID: shivam@vedanadhi.com. ORCID: 0000-0002-1998-4199.

Abstract

Objective: To technically validate SREMAN (Systematic Reviews, Evidence Synthesis & Meta-Analysis Manager), a browser-based literature screening and evidence-management application, with respect to bibliographic import, integrated PubMed retrieval, cross-database interoperability, screening-data transfer, project-state preservation, and operational performance. **Methods:** SREMAN was evaluated on October 7, 2026. Testing included a PubMed CSV file containing 6,645 records retrieved using “Leptin gene AND obesity” and a separate integrated PubMed retrieval workflow using “5-HTTLPR AND suicide,” which returned 134 records. Cross-database interoperability was assessed using CSV and RIS files from PubMed, Cochrane, ClinicalTrials.gov, and ScienceDirect. A project containing 30,963 records was used to evaluate CSV export, JSON backup and restoration, project clearing, and fresh-project import. An 18-record modified or incomplete CSV file was also tested. Functional outcomes were assessed against predefined acceptance criteria. Performance measurements were obtained using browser developer performance recordings, with three runs per operation; arithmetic means were calculated. **Results:** All tested source files were successfully imported under the validation conditions. The modified CSV was processed with unavailable information remaining blank or unavailable rather than being intentionally reconstructed. Mean execution times were 24.7 milliseconds for initial application loading, 6.57 seconds for integrated PubMed retrieval, 2.70 seconds for importing 6,645 records, 2.98 seconds for exporting 30,963 records to CSV, and 3.09 seconds for JSON backup. Project clearing required 2.39 seconds, JSON restoration required 9.81 seconds, and fresh-project import of 30,963 screening records required 21.36 seconds. The large project was successfully exported, cleared, restored from JSON, and re-imported. CSV re-import recovered information represented in the exported fields, whereas JSON restoration recovered the broader project state represented in the backup. **Conclusion:** SREMAN demonstrated functional support for large-scale bibliographic import, integrated PubMed retrieval, practical CSV/RIS interoperability with the tested sources, screening-data transfer, and project-state recovery. The performance measurements provide environment-specific evidence of operational feasibility for projects containing tens of thousands of records. However, the validation does not establish universal format compatibility, automatic duplicate detection, formal inter-reviewer reliability, or equivalence to established systematic-review methodologies. SREMAN may serve as a lightweight tool for browser-based literature screening and evidence management, with further validation warranted across additional environments and workflows. **Keywords:** literature screening; systematic review; evidence synthesis; bibliographic database; software validation.

© 2026 Vishnu Shivam. Licensed under CC BY 4.0.

1. Introduction

Evidence synthesis frequently requires identification of potentially relevant literature from one or more bibliographic databases, followed by screening, exclusion, documentation, and preservation of the records and decisions generated during the review process. Transparent documentation of the identification, screening, eligibility, and inclusion process is an important component of systematic-review methodology. The PRISMA 2020 statement provides recommendations for reporting systematic reviews, including reporting the flow of records through the review process and reasons for exclusion where applicable [1,2]. PubMed is a major biomedical literature resource maintained by the U.S. National Library of Medicine and provides access to bibliographic records and abstracts from biomedical literature. PubMed also provides mechanisms for exporting citation information, including CSV output for search results [3,4]. In practical literature-screening workflows, however, records may originate from several databases, and exported files may be supplied in different formats. SREMAN (Systematic Reviews, Evidence Synthesis & Meta-Analysis Manager) was developed as a browser-based literature screening and evidence-management application intended to manage bibliographic records locally while preserving screening information and project-level search information. The application uses IndexedDB for persistent client-side storage. IndexedDB is a browser API for persistent storage of structured data within web applications [5,6,7].

An important design characteristic of SREMAN [7] is the separation of bibliographic article records from project-level search and retrieval information. This allows the application to retain article-level screening information while also recording the searches and imports through which records entered the project. The current version does not perform automatic duplicate detection based on PMID, DOI, title, author, or other bibliographic identifiers. When duplicate or repeated records are encountered, duplicate status can instead be handled manually during

screening using the application's exclusion-reason mechanism.

The technical validation was informed by the previously preregistered SREMAN functional and feasibility evaluation protocol available on the Open Science Framework (OSF) [8].

The purpose of the present technical validation was to evaluate the principal data-management and interoperability functions of SREMAN through repeated practical workflows involving large-scale PubMed import, integrated PubMed retrieval, CSV and RIS import from multiple sources, modified-input testing, screening, export, project clearing, JSON backup and restoration, and performance measurement.

2. Statement of Significance:

Problem or Issue: Literature screening requires reliable management of heterogeneous bibliographic records, screening decisions, search provenance, and recoverable project data across multiple databases and file formats.

What Is Already Known: Bibliographic databases provide citation-export facilities, and existing evidence-synthesis tools support literature screening and review management. However, interoperability, data portability, and project-state recovery remain important practical considerations in research workflows.

What This Paper Adds: This study provides an empirical technical validation of SREMAN, a browser-based literature screening and evidence-management tool. Using files from PubMed, Cochrane, ClinicalTrials.gov, and ScienceDirect, it evaluates CSV/RIS import, integrated PubMed retrieval, screening-data export, JSON backup and restoration, and performance with projects containing up to 30,963 records.

Who Would Benefit from the New Knowledge in This Paper: Biomedical informaticians, systematic reviewers, evidence-synthesis researchers, and research software developers seeking practical approaches to bibliographic data interoperability, local evidence management, and project preservation.

3. Technical report

3.1. Application architecture

SREMAN is implemented as a browser-based application using persistent local browser

storage. Bibliographic records contain fields such as PMID, title, abstract, authors, journal, publication year, publication type, DOI, and other bibliographic information supported by the application.

Screening-related information includes the screening decision, exclusion reason, screening notes, search-source information, and first-retrieval information. The current screening workflow provides two primary decisions: Include and Exclude. A record identified as a duplicate can be manually assigned the exclusion reason “Duplicate.” Automatic bibliographic deduplication is not performed by the current version.

3.2. Progressive Web Application and Offline Operation

SREMAN is implemented as a Progressive Web App (PWA), allowing the application to be used directly through a web browser and, on supported devices and browsers, to be added to the device’s home screen in an app-like form. The PWA architecture allows the application resources and locally stored project data to remain available for continued use without requiring continuous internet connectivity.

Except for functions that require communication with an external database, particularly the integrated PubMed search and retrieval function, the principal SREMAN workflows are designed to operate using locally stored application data. These include viewing and screening imported records, recording screening decisions and exclusion reasons, searching and filtering the local project, navigating records, importing bibliographic files, exporting data, creating JSON project backups, restoring project backups, and clearing project data.

Accordingly, SREMAN can be used as a locally stored browser application for offline literature-screening and project-management activities after the required bibliographic data have been imported. Internet connectivity is required when using functions that retrieve new records from external services such as PubMed.

3.3. Validation date and software version

The validation was performed on October 7, 2026, using SREMAN version 1. The primary

desktop validation environment was an HP laptop with 4 GB RAM running Windows 11 and Microsoft Edge version 154.0.4258.62. Additional browser testing was performed on an iPhone 17 running iOS 27.0.0 using Google Chrome version 154.0.8037.55 and Safari. The principal quantitative performance measurements were obtained using the Windows 11/HP laptop/Microsoft Edge environment.

3.4. Validation datasets and source files

The validation used bibliographic files from multiple external sources together with data generated by SREMAN itself.

The principal datasets were as follows.

3.4.1. PubMed exported dataset

A PubMed search using the query: “Leptin gene AND obesity” returned 6,645 records. The results were exported directly from PubMed as a CSV file. This file was used for large-scale external bibliographic import validation.

3.4.2. Integrated SREMAN PubMed retrieval

A separate PubMed workflow was performed through SREMAN’s built-in PubMed search and retrieval functionality. The query: “5-HTTLPR AND suicide” retrieved 134 records.

This was a separate functional test from the 6,645-record exported PubMed dataset. The 134 records were retrieved through SREMAN’s integrated retrieval function rather than being imported from a manually exported PubMed CSV file. This workflow was primarily used to validate integrated PubMed retrieval and measure its execution time.

3.4.3. Cross-database files

Manual cross-database import was evaluated using the following source formats: CSV format tested in PubMed and Cochrane databases, and RIS format tested in Cochrane, ClinicalTrials.gov and ScienceDirect databases. The original files used for these tests are provided in the corresponding appendices [Appendices B-F].

3.4.4. Large project validation dataset

The combined validation workflow resulted in a SREMAN project containing 30,963 records. This project was used for large-scale export, backup, clearing, restoration, and fresh-import testing.

3.4.5. Modified/incomplete input

A CSV containing 18 records with intentionally modified or incomplete fields was used to assess how the importer handled altered input. Fields and/or columns were removed or modified before re-import.

3.5. CSV import

SREMAN provides a defined CSV field structure for bibliographic and screening information. The expected fields include, where available: Review, PMID, Title, Abstract, DOI, Decision, Search Sources, First Retrieved, Intervention/Exposure, Comparator, Outcome, Study Design, Inclusion Criteria and Exclusion Criteria. CSV files originating from other databases can be imported after their columns are mapped or modified to correspond to the fields recognized by SREMAN.

This design provides practical cross-database interoperability while recognizing that different bibliographic databases use different column names and structures. Accordingly, the CSV import mechanism is schema-based rather than dependent on a single database-specific export format.

During validation, source CSV files were imported successfully after appropriate field correspondence. Modified CSV files were also re-imported to determine how SREMAN behaved when expected information was absent or altered. Missing information was left unavailable or blank rather than being inferred or fabricated by the application.

3.6. RIS import

SREMAN implements a general RIS parser designed to accommodate standard RIS records and common variations in RIS field representation.

The parser recognizes RIS field tags and maps commonly used fields into SREMAN bibliographic fields. These include, where available, record type, PMID or PubMed identifier, registry identifier, RIS record identifier, title, abstract, journal, publication year, authors, DOI, database, publisher, URL, publication dates, article types, and other source-specific metadata.

The parser also supports repeated RIS fields, field continuation lines, and common alternative field representations. Authors may be obtained from common author tags, while title, abstract, journal, year, DOI, and other bibliographic information are obtained from corresponding RIS fields.

The current RIS importer does not automatically filter records based on PMID, DOI, title, authors, RIS identifier, or another bibliographic identifier. Each successfully parsed RIS record is retained as an independent article record. Consequently, duplicate handling remains a manual screening function in the current version.

RIS interoperability was empirically evaluated using files from Cochrane, ClinicalTrials.gov, and ScienceDirect. All tested RIS files were successfully processed and imported under the validation conditions. These results demonstrate practical compatibility with the tested RIS files. They should not be interpreted as formal conformance testing against every possible RIS implementation or vendor-specific RIS variation.

3.7. Screening functionality

SREMAN provides article-level screening decisions using Include and Exclude options.

For excluded records, predefined exclusion reasons can be assigned. These include, where applicable: wrong population; wrong intervention; wrong comparator; wrong outcome; wrong study design; wrong setting; wrong publication type; animal study; review article; conference abstract; duplicate; no usable data; and other. The “Duplicate” category is a manual screening decision rather than the output of an automatic deduplication algorithm. Free-text screening notes can also be recorded.

Records can be filtered according to screening status, including unscreened, included, and excluded records. Text-based filtering can be used to search bibliographic fields such as PMID, title, journal, or author. Previous/Next navigation supports sequential screening within the active filtered record set. Keyboard shortcuts are provided to facilitate sequential screening. When the user is not actively interacting with an input, textarea, or selection field, the Right Arrow key advances to the next article, the Left

Arrow key returns to the previous article, and the I and E keys assign Include and Exclude decisions, respectively. Keyboard shortcuts are automatically disabled while an editable form control is active to reduce unintended navigation or screening decisions during data entry.

3.8. Search provenance

SREMAN records search-related information separately from article-level screening information. Search-source and first-retrieval information can be associated with imported records, while project-level search history retains information concerning searches performed within the project.

This structure allows the application to preserve information concerning how records entered a project without requiring repeated bibliographic records to represent each search encounter. Because the current version does not perform automatic deduplication, repeated records are not automatically consolidated into a single article record. Duplicate status must instead be identified and recorded manually during screening.

3.9. Validation Workflow

Validation was performed through repeated functional and data-management cycles rather than a single import-and-export operation [Appendix L]. The validation protocol was designed to evaluate application initialization, integrated database retrieval, external bibliographic import, handling of modified input, screening and project-management functions, data export, project clearing, project restoration, and repeated processing of a large bibliographic dataset.

The principal workflow consisted of the following steps:

1. Initial application loading: Application loading was measured over three repeated runs under the specified test conditions.
2. Integrated PubMed retrieval: SREMAN's integrated PubMed search and retrieval function was evaluated using the query "5-HTTLPR AND suicide," which returned 134 records. The retrieval workflow was repeated three times for quantitative performance measurement.
3. External bibliographic import: Bibliographic records exported from PubMed, Cochrane, ClinicalTrials.gov, and ScienceDirect were imported into SREMAN using CSV and RIS formats as applicable. The principal PubMed CSV contained 6,645 records. The Cochrane CSV and RIS files each contained 571 records, the ClinicalTrials.gov RIS file contained 176 records, and the ScienceDirect RIS file contained one record.
4. Modified-input validation: A modified/incomplete CSV containing 18 records was imported to evaluate application behavior when source data were altered or incomplete.
5. Additional integrated PubMed retrieval: A separate integrated PubMed search using the query "Vishnu Shivam" returned six records and was evaluated as an additional functional retrieval test.
6. Screening and project functions: Article screening, filtering, sequential navigation, manual assignment of Include and Exclude decisions, structured exclusion reasons, screening notes, and project-management functions were evaluated. Duplicate identification, when applicable, was treated as a manual screening decision rather than automatic bibliographic deduplication.
7. Large-project export: A project containing 30,963 processed records was used to evaluate data export and project-state preservation. Article-level screening data were exported as CSV, a project-state backup was exported as JSON, and a screening/statistics summary was exported as CSV.
8. Project clearing: The local project was completely cleared, removing the locally stored project data.
9. JSON restoration: The previously generated JSON project backup was restored, and the restored project was examined to determine whether the preserved project state was recovered.
10. Second clearing and reload: The project was cleared again and the application was reloaded to establish an otherwise empty project state.

11. Fresh-project import: The large 30,963-record validation workload was processed in an otherwise empty project to evaluate import behavior without pre-existing project data.
12. Repeated performance measurement: Import, export, backup, restoration, clearing, and application-loading operations were measured using repeated runs where applicable. Performance measurements were obtained using the browser's developer performance tools. For import operations, timing began with application-side processing following file selection and ended when the corresponding completion notification was displayed; user time spent browsing for and selecting the file was excluded.
5. Project clearing: locally stored project data were removed and the application returned to an empty-project state.
6. Integrated PubMed retrieval: the requested query returned records through SREMAN's integrated retrieval workflow.
7. Modified-input handling: altered or incomplete input was processed without unsupported information being fabricated.
8. Performance: execution time was successfully measured using the defined browser performance-recording procedure.

3.11. Quantitative Validation

3.11.1. Performance measurement procedure

Performance measurements were obtained using browser developer performance tools.

For integrated PubMed retrieval, execution time represented the application-side search and record-retrieval workflow.

For CSV and RIS import, timing began with initiation of application-side processing after the user had selected the file and excluded the time spent browsing for and selecting the file. Timing continued through parsing, processing, local storage operations, retrieval/display operations, and the application's import-completion notification.

For export and backup operations, timing represented the application-side generation of the requested output. For project clearing, timing represented the application-side removal of project data. For JSON restoration, timing represented restoration of the project data into the local application state.

Each quantitative operation was measured three times under the same test conditions. Arithmetic means were calculated from the three observations.

3.11.2. Quantitative performance results

The measurements demonstrate that the tested local operations completed within tens of seconds, including operations on a project containing 30,963 records [Table 1].

The complete validation sequence, source files, screenshots, and supporting project files are provided in the accompanying appendices [Appendices A-M]. The validation included testing on an HP laptop with 4 GB RAM running Windows 11 and Microsoft Edge, as well as Safari and Chrome on iOS 27.0.0. Quantitative performance measurements reported in this study were obtained under the specified Windows 11/Microsoft Edge test environment.

3.10. Validation criteria

A validation operation was considered a 'Pass' when the observed application behavior corresponded to the predefined expected behavior for that operation.

The criteria were:

1. Import: all records contained in the tested input file were successfully processed and made available within the application without an import failure.
2. Export: the requested export file was successfully generated and contained the expected project or article-level information represented by that export format.
3. CSV restoration: information explicitly represented in the CSV was recoverable following project clearing and re-import.
4. JSON restoration: the project state represented in the JSON backup was

Table 1. Quantitative performance measurements

Operation	Test condition	Run 1	Run 2	Run 3	Mean
Initial application loading	SREMAN startup	29 ms	24 ms	21 ms	24.7 ms
Integrated PubMed search/	134 records	6.573 s	8.202 s	4.928 s	6.57 s
CSV import	6,645 records	2.381 s	2.578 s	3.155 s	2.70 s
CSV import	571 records	3.598 s	3.383 s	4.376 s	3.79 s
RIS import	571 records	3.101 s	3.516 s	3.857 s	3.49 s
RIS import	176 records	3.715 s	3.307 s	3.524 s	3.52 s
Summary CSV export	Summary output	188 ms	199 ms	195 ms	194 ms
CSV export	30,963 records	2.930 s	3.090 s	2.918 s	2.98 s
JSON project backup	30,963 records	3.178 s	3.131 s	2.957 s	3.09 s
Complete project clearing	30,963 records	2.436 s	2.746 s	1.987 s	2.39 s
JSON project restoration	30,963 records	9.474 s	12.076 s	7.894 s	9.81 s
Fresh CSV screening data import into empty project	30,963 records	20.258 s	20.212 s	23.606 s	21.36 s

The most time-consuming operation was a fresh import into an empty project, with a mean execution time of 21.36 seconds. Restoring the 30,963-record project from JSON required a mean of 9.81 seconds.

The measurements should be interpreted as application execution times under the specified hardware, operating system, browser, dataset, and test conditions rather than as universal performance benchmarks.

3.12. Functional Validation Results

3.12.1. Cross-database import

All tested source files from PubMed, Cochrane, ClinicalTrials.gov, and ScienceDirect were successfully imported under the validation conditions. The tested formats were: PubMed CSV; Cochrane CSV; Cochrane RIS; ClinicalTrials.gov RIS; and ScienceDirect RIS. The original source files are provided in Appendices B–F.

The successful import of these files demonstrates practical interoperability with the tested external databases and file formats. It does not establish compatibility with every database or every vendor-specific CSV/RIS implementation.

3.12.2. Large-scale project validation

A project containing 30,963 records was used for large-scale validation of project management functions.

The project was successfully:

1. populated with imported records;
2. screened and modified;
3. exported as CSV;
4. backed up as JSON;
5. completely cleared;
6. restored from the JSON backup;
7. cleared again;
8. reloaded; and
9. freshly populated through re-import.

These repeated operations demonstrated successful handling of a project substantially larger than the individual 6,645-record PubMed import.

3.12.3. CSV export and re-import

SREMAN provides an article-level CSV export containing the bibliographic and screening information represented by the export schema. The exported screening data were subsequently re-imported following project clearing.

Information explicitly represented within the CSV was restored following re-import. However, the CSV should not be considered a complete snapshot of the internal SREMAN project state. Project-level information not represented in the

CSV is not necessarily reconstructed by CSV re-import. This distinction is important because CSV is primarily an article-level data-interchange mechanism, whereas JSON backup is designed to preserve the broader application state.

3.12.4. JSON project backup and restoration

The JSON backup mechanism was tested using the 30,963-record project. The project was first backed up as JSON and then completely cleared. The previously generated JSON file was subsequently restored.

The restoration successfully reconstructed the project information represented in the backup, including article records, screening information, project search information, PICO-related information, database statistics, and other preserved project state. The successful restoration after complete project clearing demonstrates that JSON functions as a project-state recovery mechanism rather than merely as an alternative bibliographic export format.

3.12.5. Project clearing

The complete project-clearing function was independently evaluated using the large validation project. Following clearing, the previously stored local project information was removed and the application returned to an empty-project state. The cleared project was subsequently restored from JSON and independently repopulated through fresh import testing. This provided two distinct recovery workflows: (1) restoration from a previously generated JSON project backup; and (2) re-import of portable article-level data into an empty project.

3.12.6. Modified and incomplete CSV validation

An 18-record CSV was intentionally modified by removing or altering fields and/or columns before import. The purpose of this test was to determine whether SREMAN would generate unsupported bibliographic information when source data were incomplete. The importer processed the altered input while unavailable fields remained blank or unavailable. No unsupported information was intentionally reconstructed from absent fields. This behavior is

desirable for evidence-management workflows because missing source information should remain identifiable as missing rather than being silently inferred.

3.12.7. Supporting Evidence and Appendices

The validation was supported by original bibliographic export files, project-state files, screening-data exports, a PRISMA-style screening summary, modified input files, and screenshots documenting the principal application states and validation workflows. These materials were retained as supporting evidence of the validation procedures and outcomes reported in this technical report (Appendices A–M).

The original files and supporting screenshots provide an auditable record of the source data, imported records, exported data, project state, and application states used during validation. The appendices are intended to supplement, rather than replace, the quantitative results and methodological description provided in the main report.

3.13. Validation Inventory

Table 2 summarizes the principal source files and application-generated files used during validation, including their source, format, record count where applicable, validation purpose, and outcome.

4. Discussion

The present validation demonstrates that SREMAN can support a local browser-based workflow for bibliographic import, screening, project management, export, backup, restoration, and repeated data-management operations.

A principal strength of the validation is that it was conducted using real bibliographic files from multiple external sources rather than relying exclusively on synthetic test data. The principal PubMed dataset contained 6,645 records and was imported from a CSV file exported directly from PubMed. Additional source files were obtained from Cochrane, ClinicalTrials.gov, and ScienceDirect and were tested in CSV or RIS format as applicable.

The validation also distinguished between two different PubMed functions. The 6,645-record dataset represented external CSV import from

PubMed, whereas the 134-record test represented database retrieval involve different application SREMAN's integrated PubMed search and workflows. retrieval functionality. This distinction is important because external file import and direct

Table 2. File-level validation inventory

File/workflow	Source	Format	Records	Validation purpose	Outcome	Supporting appendix
PubMed export	PubMed	CSV	6,645	Large-scale external import	Pass	Appendix B
Cochrane export	Cochrane	CSV	571	Cross-database CSV import	Pass	Appendix E
Cochrane export	Cochrane	RIS	571	RIS import	Pass	Appendix D
ClinicalTrials.gov export	ClinicalTrials.gov	RIS	176	RIS import	Pass	Appendix C
ScienceDirect export	ScienceDirect	RIS	1	RIS import	Pass	Appendix F
Integrated PubMed retrieval	SREMAN/PubMed	Search retrieval	134	Integrated retrieval and performance validation	Pass	Appendix K, L
Integrated PubMed retrieval	SREMAN/PubMed	Search retrieval	6	Additional retrieval validation	Pass	Appendix K, L
Modified input	-	CSV	18	Modified/incomplete input handling	Pass	Appendix J, L
Screening-data export	SREMAN	CSV	30,963	Article-level data export	Pass	Appendix I
PRISMA-style screening summary	SREMAN	CSV	30,963	Screening-summary export	Pass	Appendix H
Project backup	SREMAN	JSON	30,963	Project-state preservation	Pass	Appendix G
Project restoration	SREMAN	JSON	30,963	Recovery after project clearing	Pass	Appendix G, L

The cross-database testing demonstrated maps multiple bibliographic fields into the practical interoperability with the specific source application's internal article representation. The files evaluated. SREMAN's CSV import current implementation retains each parsed RIS mechanism allows users to align source-specific record and does not perform automatic duplicate column headings with the fields expected by the filtering.

application. This approach permits records from a variety of databases with different CSV approaches that rely on automatic bibliographic structures to be imported following appropriate deduplication. In the current SREMAN version, field mapping or column alignment. However, duplicate status is instead a manual screening compatibility depends on the availability and decision. This avoids making unsupported adequacy of the source data for mapping to the assumptions about record identity when fields supported by SREMAN. identifiers are absent, inconsistent, or

RIS support was also demonstrated across represented differently across databases. multiple sources. The general RIS parser However, it also places additional responsibility recognizes common RIS field structures and

on the evaluator to identify duplicate records during screening.

The 30,963-record project provided the most extensive validation of the application's local project-management capabilities. The project was successfully exported, backed up, cleared, restored, and freshly re-imported. The ability to perform these operations repeatedly provides evidence that the application can manage a substantially larger project than the individual 6,645-record PubMed import used in the principal external-import test.

The distinction between CSV export and JSON backup is technically important. CSV is useful as a portable, human-readable article-level interchange format. However, it does not necessarily represent every internal state variable maintained by a browser application. Consequently, CSV re-import can restore the information represented by the CSV while not recreating project-level information that was never encoded in the CSV.

JSON backup serves a different purpose. It preserves the broader SREMAN project state represented in the backup and can therefore be used as a recovery mechanism following complete project clearing.

The performance measurements provide additional quantitative support for the reported functionality. The 30,963-record fresh-import workflow required a mean of 21.36 seconds under the specified desktop test conditions. JSON restoration of the same project required a mean of 9.81 seconds, while CSV export and JSON backup required mean times of 2.98 and 3.09 seconds, respectively.

These measurements should not be interpreted as formal software benchmarks. Performance depends on hardware, browser implementation, storage conditions, dataset structure, and application state. The measurements nevertheless provide reproducible observations under the stated validation conditions and are substantially more informative than qualitative statements that operations occurred within seconds.

The modified CSV test further demonstrated that SREMAN did not intentionally reconstruct absent information. When fields were removed

or altered, unavailable information remained unavailable rather than being fabricated. This behavior is appropriate for bibliographic evidence management, where preservation of source fidelity is more important than artificially increasing apparent data completeness.

An additional practical feature of SREMAN is its Progressive Web App (PWA) architecture. The application can be used through a browser and, on supported platforms, added to the device home screen to provide an app-like experience. Because the principal screening and project-management functions operate on locally stored application data, these functions can continue to be used without continuous internet connectivity after the required bibliographic records have been imported. External database retrieval functions, such as the integrated PubMed search, require an active internet connection. This architecture may be particularly useful for researchers who need to perform screening and project management in environments with intermittent or limited network connectivity.

4.1. Limitations:

The validation also has several limitations.

First, the current version does not provide automatic bibliographic deduplication. Duplicate records therefore require manual identification and classification during screening. Future versions may implement identifier-assisted or bibliographic similarity-based duplicate detection.

Second, the CSV workflow depends on field correspondence. Although this provides flexibility because users can map source-specific columns to SREMAN fields, it also means that incorrectly mapped CSV columns may produce incomplete or incorrectly classified information.

Third, the RIS validation demonstrates compatibility with the specific RIS files tested but does not establish universal compatibility with every RIS implementation. RIS is implemented differently across bibliographic platforms, and additional validation against a broader corpus of RIS variants would be required for a formal interoperability claim.

Fourth, the validation was not designed as a formal comparative benchmark against

commercial or open-source systematic-review platforms.

Fifth, the present application does not provide synchronized multi-reviewer adjudication. It is therefore best suited to individual or locally managed screening workflows rather than being characterized as a full collaborative systematic-review platform.

Sixth, the present validation did not evaluate a dedicated full-text assessment workflow.

Seventh, browser testing was limited to the environments described in this report. Testing Safari, Chrome, and Microsoft Edge under the stated conditions does not establish compatibility with every browser, operating system, private-browsing configuration, device, storage policy, or future browser release.

Finally, because the application relies on local browser storage, users should maintain regular project backups. Browser-managed storage may be removed by users or become unavailable because of browser storage policies or other environmental conditions [5,6].

Future development could include automated or assisted duplicate detection, expanded identifier matching, configurable CSV field mapping, source-specific import profiles, formal RIS conformance testing, full-text assessment, multi-reviewer screening, blinded independent review, adjudication workflows, synchronized project storage, and formal usability and performance benchmarking.

5. Conclusions

This technical validation demonstrated that SREMAN can support a local browser-based workflow for literature screening and evidence management involving external bibliographic import, integrated PubMed retrieval, screening, cross-database CSV and RIS interoperability, data export, project clearing, JSON backup, and project restoration.

The validation incorporated real-world bibliographic files from PubMed, Cochrane, ClinicalTrials.gov, and ScienceDirect. A 6,645-record PubMed CSV was successfully imported, while a separate 134-record PubMed search was successfully retrieved through SREMAN's integrated PubMed search function. A larger

30,963-record project was successfully exported, backed up, cleared, restored, and freshly re-imported.

Quantitative performance testing demonstrated mean execution times of 2.70 seconds for import of the 6,645-record PubMed CSV, 2.98 seconds for CSV export of 30,963 records, 3.09 seconds for JSON backup, 9.81 seconds for JSON restoration, and 21.36 seconds for fresh import of 30,963 records under the specified desktop test conditions.

The validation supports SREMAN as a lightweight browser-based screening and evidence-management application capable of handling large local bibliographic projects and preserving project state through portable JSON backup. The findings do not establish universal compatibility with all bibliographic databases or file formats, and the current version does not provide automatic duplicate detection, synchronized multi-reviewer adjudication, or a dedicated full-text assessment workflow.

Accordingly, SREMAN should be considered a technical tool supporting literature screening and evidence management rather than a replacement for systematic-review methodology, formal duplicate-review procedures, or PRISMA reporting requirements.

Declarations:

Conflict of Interest:

Vishnu Shivam is the developer of SREMAN (Systematic Reviews, Evidence Synthesis & Meta-Analysis Manager) and the owner and operator of Vedanadhi. The author therefore has a direct developer and ownership-related interest in the software evaluated in this study. SREMAN is currently provided without charge, and the author receives no financial compensation or commercial revenue from its use. This relationship is disclosed to ensure transparency. The validation procedures and predefined evaluation criteria were applied to assess the documented functionality of the software.

Funding: None

Ethics statement: Not applicable

Software Availability:

SREMAN is available at <https://sreman.com>.

Data and Supporting Materials:

The original bibliographic source files, modified test files, project backup files, screening-data export, and supporting screenshots used for the technical validation are provided as appendices or supplementary materials accompanying this manuscript, subject to applicable database-provider terms and copyright restrictions.

Acknowledgments:

The author acknowledges the National Library of Medicine and PubMed for providing public access to biomedical bibliographic records and programmatic retrieval and citation-export functionality used in the technical validation of SREMAN. The author also acknowledges the bibliographic database providers whose publicly available export files were used to evaluate cross-database import functionality.

The author also acknowledges OpenAI's ChatGPT for assistance during manuscript preparation, including language and grammar editing, formatting, organization of the manuscript, and assistance with code development and refinement. The author independently reviewed, verified, and accepted responsibility for the final manuscript content, technical implementation, validation procedures, interpretation of results, and conclusions.

Supplementary information:

Available in the online version. SREMAN.com

Appendix A

Detailed technical validation report, including validation procedures, acceptance criteria, quantitative measurements, validation matrices, and interpretation.

Appendix B

Original PubMed CSV export containing 6,645 records generated from the query "Leptin gene AND obesity."

Appendix C

Original RIS file exported from ClinicalTrials.gov containing 176 records.

Appendix D

Original RIS file exported from Cochrane containing 571 records.

Appendix E

Original CSV file exported from Cochrane containing 571 records.

Appendix F

Original RIS file exported from ScienceDirect containing one record.

Appendix G

JSON project backup corresponding to the 30,963-record project used for restoration validation.

Appendix H

PRISMA-style screening summary exported as CSV from the 30,963-record project.

Appendix I

Article-level screening-data CSV corresponding to the 30,963-record project.

Appendix J

Modified/malformed 18-record CSV used for input-validation testing.

Appendix K

Project JSON data corresponding to the repeated integrated PubMed-search validation.

Appendix L: Supplementary Screenshots Documenting Application States and Technical Validation Workflows

This appendix presents supplementary screenshots documenting the principal application states and technical validation workflows of SREMAN. The figures illustrate application initialization, PubMed record retrieval, cross-database bibliographic import, article screening, CSV export and import, JSON backup and restoration, validation of modified or malformed CSV files, and repeated PubMed searches. The validation workflows also document the handling of a large project comprising 30,963 bibliographic records.

Appendix M: Large-Project Dataset Composition and Import Inventory

The 30,963-record project was assembled through repeated file-import operations, integrated PubMed retrievals, and a modified 18-record CSV import rather than through a single import file. The inventory below identifies the documented source files and retrievals, their record counts, and their contributions to the project. Repeated imports and retrievals are listed separately from unique source files because repeated operations can increase the project record count even when the underlying source file or search query is unchanged.

The modified or incomplete CSV contained 18 records and was imported at the end of the documented import sequence, immediately before the six records retrieved through the integrated PubMed search for “Vishnu Shivam.” The 18 records were therefore included as a distinct import operation rather than being treated as an unexplained remainder.

The documented operations account for all 30,963 records in the final project. The table

Table M1. Import and retrieval operations contributing to the large validation project

Source or operation	Format or function	Records per operation	Number of operations	Total records contributed
PubMed export: “Leptin gene AND obesity”	CSV import	6,645	4	26,580
Cochrane export	CSV import	571	3	1,713
Cochrane export	RIS import	571	3	1,713
ClinicalTrials.gov export	RIS import	176	3	528
ScienceDirect export	RIS import	1	3	3
Integrated PubMed search: “5-HTTLPR AND suicide”	PubMed retrieval	134	3	402
Modified or incomplete CSV	CSV import	18	1	18
Integrated PubMed search: “Vishnu Shivam”	PubMed retrieval	6	1	6
Total documented operations	—	—	21	30,963

References:

1. Page MJ, McKenzie JE, Bossuyt PM, et al. The PRISMA 2020 statement: an updated guideline for reporting systematic reviews. *BMJ*. 2021;372:n71. <https://doi.org/10.1136/bmj.n71>.
2. Page MJ, Moher D, Bossuyt PM, et al. PRISMA 2020 explanation and elaboration: updated guidance and exemplars for reporting systematic reviews. *BMJ*. 2021;372:n160. <https://doi.org/10.1136/bmj.n160>.
3. National Library of Medicine. PubMed. National Center for Biotechnology Information. Accessed October 4, 2026. <https://pubmed.ncbi.nlm.nih.gov/>.
4. National Library of Medicine. Download PubMed data. National Center for Biotechnology Information. Accessed October 4, 2026. <https://pubmed.ncbi.nlm.nih.gov/download/>.
5. Mozilla. IndexedDB API. MDN Web Docs. Accessed October 4, 2026. https://developer.mozilla.org/en-US/docs/Web/API/IndexedDB_API.
6. Mozilla. Using IndexedDB. MDN Web Docs. Accessed October 4, 2026. <https://>

distinguishes the number of records contained in each source file or returned by each retrieval from the number of records contributed by repeated operations. The resulting project count describes the number of records in the assembled project; it does not represent the number of unique bibliographic publications because automatic deduplication was not performed.

developer.mozilla.org/en-US/docs/Web/API/IndexedDB_API/Using_IndexedDB.

7. SREMAN: Systematic Reviews, Evidence Synthesis & Meta-Analysis Manager. Accessed October 4, 2026. <https://www.sreman.com/>.
8. Shivam V. Functional and Feasibility Evaluation of SREMAN: A Browser-Based Tool for Systematic Review Screening [Internet]. OSF; 2026. Available from: <https://osf.io/9nf7d>.

Citation:

Shivam V. Technical Validation of SREMAN: A Browser-Based Systematic Review and Evidence Management Tool for Cross-Database Interoperability, Screening-Data Preservation, and Project-State Restoration. Vedanadhi. 2026. Available from: <https://sreman.com/tr.html>

First published online: 20 October 2026

Version: 1.0

Peer review: None