THE JOURNAL OF
MEDICINE, LAW & PUBLIC HEALTH

Vol 6, No 2. 2026
Original Articles

The Newfoundland and Labrador Geriatric Health Index (NLGHI): Design, Implementation, and Clinical Workflow-Oriented Features for Community Geriatrics

Mirza Niaz Zaman Elin[1*],

[1] *AMAL Youth and Family Centre, St John's, Canada*

Summary

Background: Primary care and community-based settings often face time limitations that make detailed, repeatable geriatric assessment difficult. There is a need for a practical, clinic-oriented tool that supports longitudinal tracking while remaining easy to interpret, transparent, and straightforward to use in routine workflows.
Aims: This paper introduces NLGHI, a desktop-based application designed to support structured multi-domain data entry, automated composite scoring, longitudinal visualization, rapid report generation, and basic administrative functions within a single system.
Methods: The application was developed in Python using a Qt-based interface, with established libraries such as NumPy and Matplotlib supporting computation and visualization. It records impairment levels across 27 predefined clinical and social domains and calculates a normalized composite score for each visit based on fixed weightings. Data are stored locally in JSON format to allow portability and direct inspection without dependence on external systems. Additional components include a symptom-lexicon module for advisory input and a patient workspace that supports notes, attachments, follow-ups, and timeline exports.
Results: Implementation produces consistent scoring outputs and clear visual summaries, including trend graphs, across visits. Reports can be generated quickly, and patient records remain auditable within a single workstation. Basic validation checks help identify inconsistencies during data entry, while dashboards allow clinicians to review longitudinal changes without requiring server infrastructure.
Conclusion: NLGHI offers a transparent and adaptable approach to geriatric data capture in small-scale clinical environments. It is intended to complement, rather than replace, established tools such as the Clinical Frailty Scale, Charlson and Elixhauser indices, and the Katz ADL, providing a practical option for ongoing monitoring in primary care and community settings.

Keywords: Activities of Daily Living, Aged, Comorbidity, Electronic Health Records, Frailty, Geriatric Assesment, Health Status Indicators, Longitudinal Studies, Medical Informatics Applications, Primary Health Care.

Received: 2025-09-14 | Accepted: 2026-03-19 | Published: 2026-04-13
DOI: 10.52609/jmlph.v6i2.266
*Corresponding author: niazmirza111@gmail.com

INTRODUCTION

Population ageing in Atlantic Canada and elsewhere continues to increase the clinical and administrative demands on primary care physicians, nurse practitioners, and allied health teams [1]. In settings without enterprise electronic medical records (EMR) or where EMR customization is constrained, the burden of capturing structured, longitudinal geriatric information often devolves to ad hoc spreadsheets or free-text notes. Such pattern complicates trend detection, auditability, and standardized communication across caregivers.

The Newfoundland and Labrador Geriatric Health Index (NLGHI) was conceived as a desktop application to support (i) consistent capture of impairment intensities across multiple geriatric domains; (ii) automatic computation of a normalized composite index suitable for at-a-glance interpretation; (iii) longitudinal visualization to display change; and (iv) export of concise summaries for documentation and hand-off. The implementation emphasizes transparency by ensuring plain local storage, human-readable exports, and a minimal dependency chain leveraging well-documented libraries for operational components and graphics [2–4]. JavaScript Object Notation (JSON) was chosen to ease local backup and to enable inspection without proprietary tooling [5]. Numerical and visualization routines are implemented with mature packages from the scientific Python programming ecosystem, which are both widely cited and actively maintained [2–4].

The primary design goal was not to supplant validated geriatric scales used in research or specialist clinics. Instead, NLGHI was built to complement routine visits by recording impairment intensities in a manner that can be repeated and audited, thereby simplifying temporal comparisons and care coordination. The tool's conceptual positioning relative to established measures such as the Clinical Frailty Scale (CFS) [6], Charlson Comorbidity Index (CCI) [7], Elixhauser comorbidity indicators [8], and the Katz Index of Independence in Activities of Daily Living (ADL) [9] is discussed below.

METHODS

Related Work

Several indices are widely employed in geriatric practice and outcomes research. The Clinical Frailty Scale (CFS) provides an ordinal classification of frailty and has been shown to predict mortality and institutionalization [6]. The Charlson Comorbidity Index (CCI) assigns weighted scores to comorbid conditions to predict mortality risk [7]. The Elixhauser comorbidity set comprises indicators derived from administrative data and has been adapted to yield a single numeric score in later work [8]. Functional status is often summarized using the Katz ADL, which captures independence in basic activities of daily living [9]. These instruments serve distinct purposes and contexts; consequently, heterogeneous instruments are often combined in comprehensive geriatric assessment.

NLGHI does not attempt to reproduce the psychometrics or longitudinal validation of those instruments. Instead, a practical objective guided the design: a clinic-adapted, low-friction recording platform that imposes consistent structure across multiple biomedical and social domains, returns a normalized composite, and allows quick inspection of trajectories. Comparison with established measures is therefore conceptual: While the NLGHI score provides a granular internal metric for impairment intensity and distribution, the CFS, CCI/Elixhauser, and Katz ADL indices offer established benchmarks for external clinical validity. Any cross-walking between NLGHI and those instruments would require targeted empirical work.

Overview of the Application:

Scope and workflow

The NLGHI application supports a physician- or clinic-operated workflow in which a patient is identified by a medical care plan (MCP) number, and visit-level impairment intensities are selected for 27 predefined domains encompassing cardiovascular, respiratory, neurological, musculoskeletal/physical trauma, renal, hepatic, gastrointestinal, dermatological, urogenital/reproductive, oncological, haematological, genetic, endocrine, immunodeficiency, nutritional deficiency, autoimmune, ophthalmic, otolaryngological, psychiatric/psychological/mental/behavioural, oral/dental, disability, dependence on supportive aids, social well-being, economic well-being, abuse/neglect, risk factors, and a general symptoms bin [Figures 1, 2]. These health domains should not be viewed as

fixed metrics; they are context-dependent and designed to be selected according to the specific objectives of statistical studies. The corresponding weightings are based on severity levels, where 5 indicates the greatest possible severity and 1 indicates the lowest [Table 1].

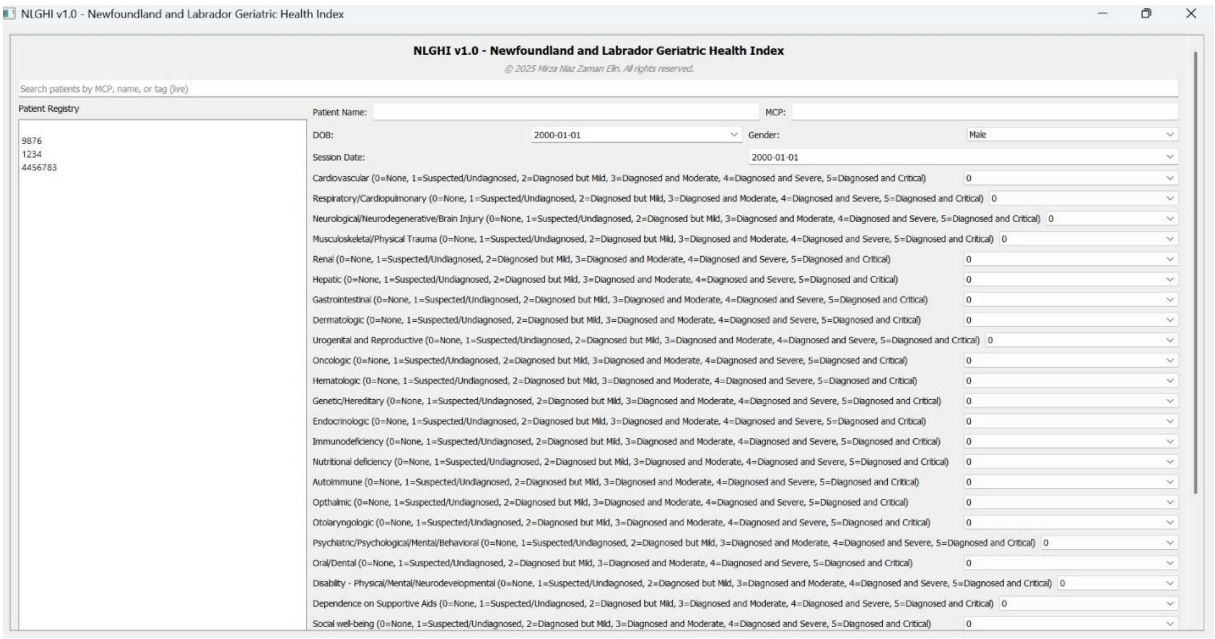

Figure 1. Screenshot of NLGHI v1.0 Data-Template

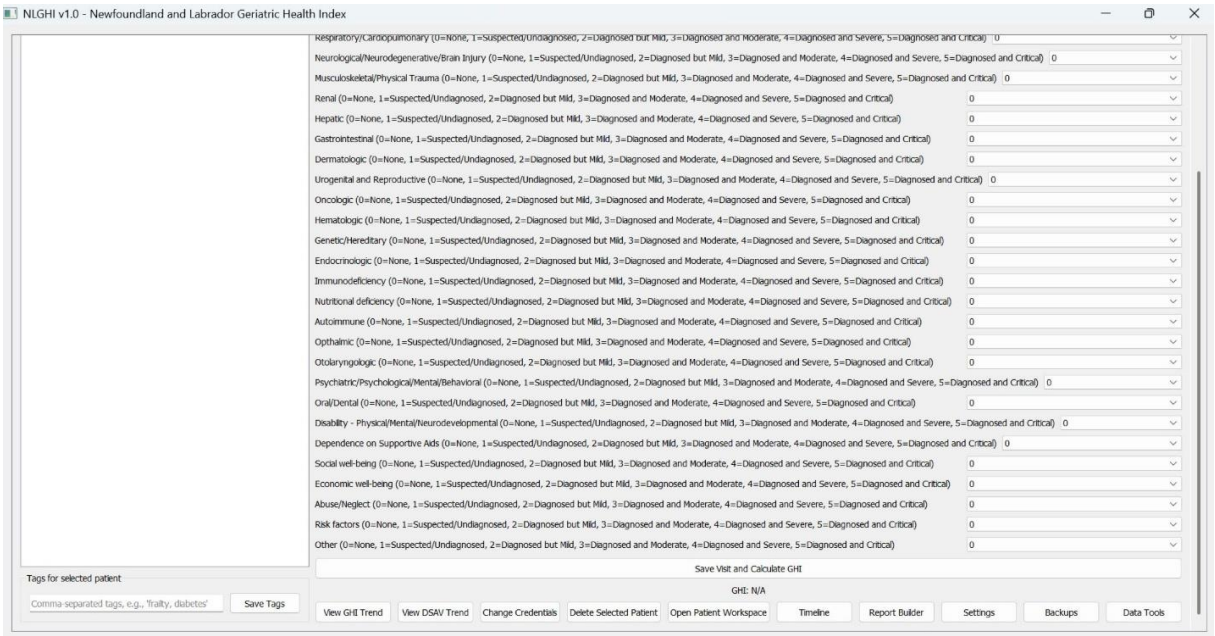

Figure 2. Screenshot of NLGHI v1.0 Automation Features

Table 1. Domains and corresponding weightings

Domain	**Weight**
Cardiovascular	5
Respiratory/Cardiopulmonary	5
Neurological/Neurodegenerative/Brain Injury	5
Musculoskeletal/Physical Trauma	4
Renal	4
Hepatic	4
Gastrointestinal	4

Dermatological	3
Urogenital and Reproductive	3
Oncological	5
Haematological	4
Genetic/Hereditary	1
Endocrinological	2
Immunodeficiency	2
Nutritional deficiency	2
Autoimmune	2
Ophthalmic	2
Otolaryngological	2
Psychiatric/Psychological/Mental/Behavioural	2
Oral/Dental	2
Disability – Physical/Mental/Neurodevelopmental	1
Dependence on Supportive Aids	1
Social well-being	1
Economic well-being	5
Abuse/Neglect	3
Risk factors	1
Other	1

Each domain selection yields a domain-specific severity value (0–5) and a domain-specific DSAV (domain severity aggregated value), which is the severity multiplied by a weighting reflecting relative salience. The total of the DSAVs is normalized to produce the Geriatric Health Index (GHI) for that visit. A report builder generates brief textual summaries, and a timeline export captures events across history, notes, and follow-ups. A symptom analyser zinterprets the data and, accordingly, provides domain suggestions for documentation support. The longitudinal monitoring report provides graphical representations of changes in DSAV and GHI overtime [Figures 3,4].

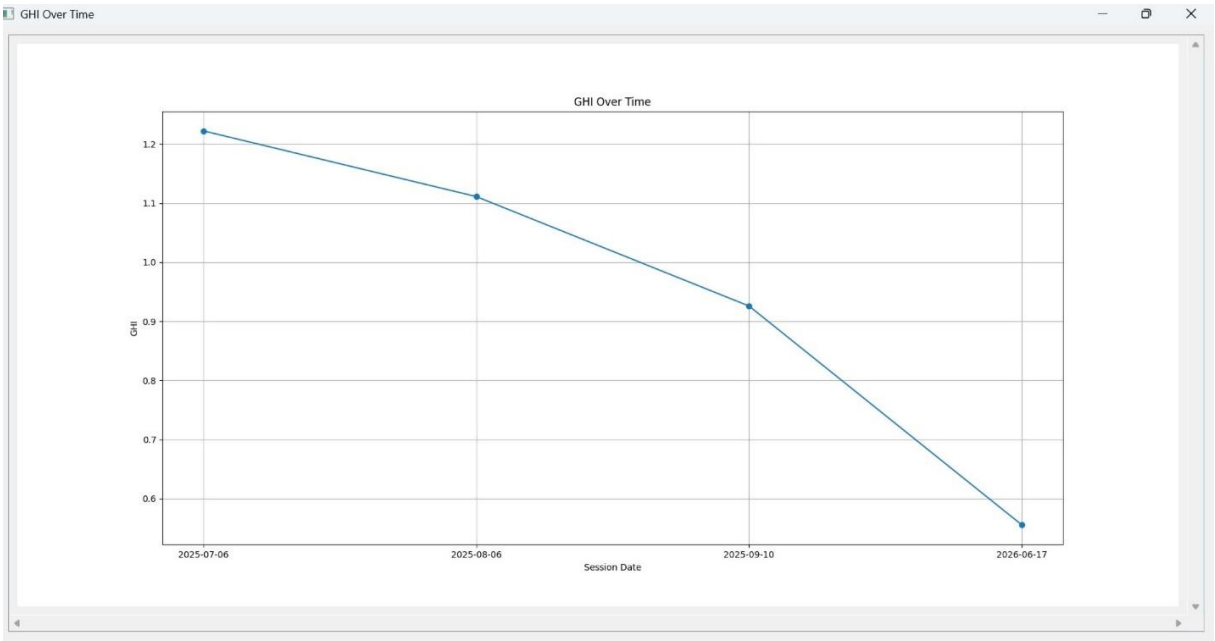

Figure 3. GHI Graphical Report Auto-Generated by NLGHI v1.0

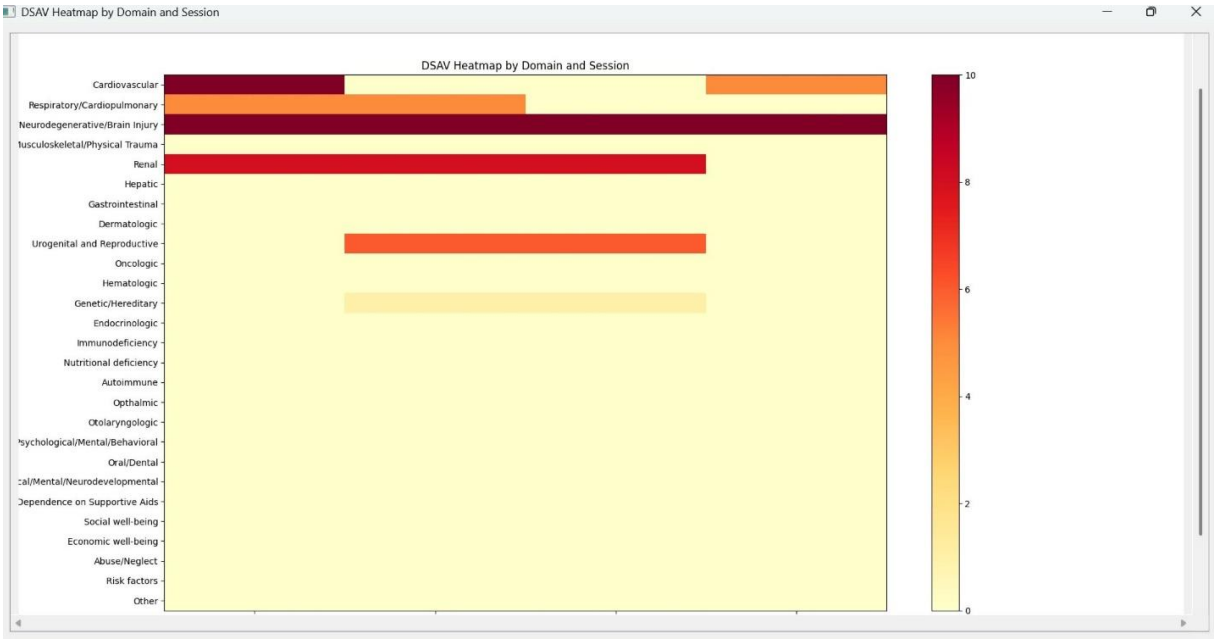

Figure 4. DSAV Graphical Report Auto-Generated by NLGHI v1.0

Implementation stack

The application is written in Python and implemented with the Qt widget toolkit for the graphical interface. Numerical arrays and plotting are provided by NumPy and Matplotlib, respectively, which are standard within the scientific Python ecosystem [2–4]. Data are stored in JSON format in a single file (default nlghi_patient_data.json), and settings, credentials (username/password), and audit logs are stored in separate local files for modularity [5]. The deliberate choice of JSON as the storage layer enables human readability, ad hoc backup, and non-proprietary migration.

Intended use and disclaimers

The application is intended for advisory and documentation support. The presence of a symptom analyser is now common in clinical software, but NLGHI's lexicon-based module is intentionally conservative; its outputs are descriptive and not intended to constitute decision support.

Data Model and Scoring:

Patient record structure

Each patient is indexed by MCP, including their demographics, visit records, history entries, notes, future references, attachments, and saved symptom snapshots. The record format is compact and human-readable. A typical visit record contains:

(A) Timestamp (save event time);
(B) Session date (visit date);
(C) Impairments: 27-length integer vector on a 0–5 scale;
(D) DSAVs: 27-length integer vector computed as severity × domain weighting;
(E) GHI: normalized scalar score (sum of DSAVs divided by 27).

A modular structure is maintained across history, notes, and attachments to facilitate exports (plain text, markdown, CSV for records). A small audit log captures major operations.

Domain weightings and normalization

The 27 domains are assigned integer weightings representing relative burden or salience. The DSAV vector is calculated component-wise as the product of impairment level and domain weighting, and the GHI is computed as a normalized sum of DSAVs divided by 27, producing a scalar that increases with overall impairment burden. The computation is straightforward and computationally lightweight; it is emphasized that no clinical threshold measurement or risk prediction model is embedded in the score. The composite is intended for within-system tracking and quick communication across team members, not as a substitute for validated prognostic indices.

Consistency checks

A Data Tools dialog performs internal validation by verifying vector lengths (27 domains), re-computing GHI from stored DSAVs, and flagging mismatches. The existence of automated checks

reduces the risk of data drift if domain definitions evolve or if edits occur outside the graphical user interface (GUI). A summary of findings (e.g., number of patients, number of issues) is presented in a read-only pane to facilitate routine housekeeping.

User Interface and Core Functions:

Patient registry and search

The left-hand pane provides a live-filtered registry by MCP, name, or tag. Tagging supports ad hoc cohort organization (e.g., "frailty," "diabetes," "home visit") and can be updated in-place. The filter is case-insensitive and scans both identifiers and tags. This functionality supports quick retrieval when the registry becomes large.

Visit capture and score display

A structured form presents the 27 domains with a clear ordinal scale for impairment. On saving, DSAVs are computed and the GHI is displayed immediately. The choice of an ordinal scale encourages conservative, category-level judgments appropriate for time-limited visits. The GHI's reproducibility derives from fixed domain definitions and weightings; drift is minimized by consistent labelling.

Longitudinal visualization

Two visualizations are embedded. First, a GHI line chart displays evolution over session dates. Second, a DSAV heatmap displays domain-by-session DSAV magnitudes, facilitating pattern recognition across visits (e.g., progressive increases in cardiopulmonary and renal DSAVs). The heat-map leverages matrix plotting with colour-bars and tick labelling handled by Matplotlib routines [4].

Patient workspace

A separate workspace supports:

(A) History entries with timestamps and editable bodies;
(B) Symptom checker: a lexicon-based analyser that extracts matched keywords and proposes domain votes;
(C) Notes with an option to attach to the latest visit;
(D) Future references (e.g., follow-ups) with due dates and status;
(E) Attachments (file paths), with open/delete controls.

A timeline dialog compiles events chronologically and exports a Markdown summary. A report builder produces concise textual summaries (latest visit or lifetime).

Authentication, settings, and audit

A lightweight login is included. On initial launch, the system skips the login screen to allow users to establish their first set of administrative credentials. Settings cover theme, backup behaviour, backup retention, and export directory. An audit helper writes time-stamped entries to a log, recording salient operations (e.g., writes, backups, restores). The objective is basic accountability in small-clinic environments.

Symptom Lexicon and Advisory Suggestions:

Construction

A simple dictionary of symptom phrases is mapped to one or more domains. The analyser lowercases input text and scans for phrase inclusion, accumulating votes by domain and returning matched keywords and a ranked suggestion list. The module is intentionally conservative—an advisory feature for documentation and triage time-saving, rather than a diagnostic engine.

Rationale and limitations

Lexicon-based matching was chosen for predictability and the ability to analyse and interpret domain-specific data in a clinical setting, avoiding the opacity of black-box models. The trade-off is limited recall/precision in free text; however, false positives are transparent because matched phrases are shown. A natural extension would involve synonym sets and a tokenizer to manage punctuation and spacing more robustly. Such extensions should remain explainable and should preserve the current transparency.

Data Management, Provenance, and Safety

Storage and formats

Patient data are stored in a single JSON file; settings, credentials, and audit logs are stored separately. This separation supports least privilege for operational tasks and simplifies backup workflows. The choice of JSON aligns with portability and with long-term accessibility considerations; the format is standardized via RFC 8259 with a persistent identifier [5].

Backup and restore

An automatic pre-write backup is performed by default, with a retention limit. A backup dialog permits manual creation and restoration, and the audit log records these operations. Backup files are time-stamped, enabling quick rollbacks in the event of data corruption or accidental edits.

Security considerations

Credentials are stored in a local JSON file with no cryptographic hardening by design; deployment is therefore intended for single-workstation, small-team environments without shared network exposure. On multi-user systems or where regulatory constraints apply, an external secret store and disk encryption are recommended; these are deployment considerations outside the scope of the v1.0 codebase.

Export and interoperation

Exports include plain text and Markdown summaries, and CSV for records. Human-readable exports allow quick stitching into letters or EMR inboxes. Attachments are stored by reference (file paths) to avoid bloat and to preserve file ownership.

RESULTS

Positioning Among Established Geriatric-Measures

The normalized NLGHI score—derived from domain-weighted impairment levels—should be interpreted as a program-internal burden index for quick longitudinal comparison. It is not calibrated against CFS [6], CCI [7], Elixhauser [8], or Katz ADL [9]; the evidence base for those instruments is extensive and tied to specific outcomes such as mortality, readmission, and functional independence. If cross-instrument alignment is desired (for example, mapping high NLGHI to CFS ≥ 5), empirical calibration would be required using a cohort with parallel measurement. Such work would need institutional review board (IRB) oversight and prospective or retrospective data with linkage across instruments.

The present system has been designed to coexist with established scales in practice. A typical workflow could involve completion of Katz ADL or CFS during a comprehensive visit, while NLGHI captures a multi-domain snapshot during the same visit and on follow-ups, yielding a trend line for in-house monitoring. Conceptually, CCI/Elixhauser focus on comorbidity burden, while NLGHI captures a broader pattern that includes social and support factors. The combination can support case conferences in community geriatrics without imposing analytic overhead.

Validation Utilities and Quality Assurance

Internal validation is provided in the Data Tools dialog. This utility checks that impairment and DSAV vectors have the correct length (27), recomputes GHI, and flags mismatches beyond a tight tolerance. These safeguards address two risks: (i) domain-set evolution without synchronized code changes, and (ii) manual edits to JSON. A structured message is displayed to guide corrective actions. Exported reports include precisely what was recorded at the time of generation, and the audit log provides a minimal chain of custody within a single-workstation environment.

DISCUSSION

Design Rationale: Lightweight and Local-First

Small practices often cannot absorb the operational complexity of client–server deployments. Accordingly, a local-first architecture was chosen, with a single executable and local data files. This stance trades multi-user concurrency for simplicity, speed, and reduced failure modes.

Transparency and Analysis Capabilities

A principal constraint was that every stored field should be interpretable by a clinician with modest technical skill; hence the use of plain JSON and human-readable exports. The symptom analyser reveals its matched phrases and domain vote counts; the GHI calculation is a single line and is re-computable by inspection. Such design should reduce resistance in settings where proprietary formats raise concerns.

Familiar Dependencies

Reliance on NumPy and Matplotlib reflects a preference for libraries with established governance, extensive documentation, and stable application programming interfaces (APIs) [2–4]. These choices simplify maintenance and allow consistent behaviour across platforms.

Example Use Cases: Routine follow-up

A patient previously flagged with moderate cardiopulmonary impairment presents for a follow-up. New DSAV entries are recorded, and the updated GHI is compared to the prior trend; the DSAV

heat-map reveals incremental increases across respiratory and cardiovascular rows. Notes and future references are updated to reflect medication adjustments and a planned pulmonary rehabilitation referral. The report builder compiles a summary for the family physician and home-care coordinator.

Example Use Cases: Case review meeting

For a complex patient requiring multidisciplinary input, the timeline export is generated for a case conference. DSAV heat-maps over three visits illustrate stabilization except for social and economic stressors, leading to non-clinical interventions (home support, income assistance). Attachments (e.g., home safety assessment) remain accessible via file path.

Example Use Cases: Data hygiene

Before quarterly reporting, the Data Tools dialog is run to ensure no DSAV or impairment vector mismatches exist. Any flagged entries are corrected and re-saved. Backups are made and older ones pruned automatically to the configured retention. Audit entries provide a record of these operations.

Ethical, Privacy, and Safety Considerations

The application stores personally identifiable information and is therefore subject to local privacy regulations. Disk encryption and strict device access control are recommended. Because the symptom analyser is lexicon-based and advisory, no clinical decision support claims are made. The GHI has not been validated as a prognostic instrument and should not be used to allocate or deny services. If research use is intended, ethical review and consent procedures should be followed to align with best practices.

Feasibility

The domains and weightings are customizable; in clinical settings they are purposed for use in baseline establishment and longitudinal health outcome monitoring. Therefore, accuracy, sensitivity, and specificity validation is only possible in the context of practical applications in real clinical settings. However, as a data collection, analysis, and longitudinal outcome monitoring framework, the model remains relevant as far as is feasibility is concerned.

Limitations

The application is a single-node system and does not support synchronization across devices. Authentication is lightweight and not suitable for adversarial environments. The symptom analyser is limited to phrase inclusion, and therefore does not handle spelling variants and complex negations. The 27-domain scheme and weightings reflect expert-informed pragmatism, not a formally derived metric. Visualization is focused on DSAV heatmaps and GHI lines; additional charts (e.g., radar plots) were excluded to maintain simplicity. There is no native import from EMR systems; however, the plain JSON structure enables ad hoc scripting if needed.

Future Work

Priority enhancements include: (i) negation handling and simple lemmatization in the symptom analyser; (ii) a pluggable storage layer to support SQLite while preserving JSON export; (iii) optional role-based access for small teams; (iv) cohort views (aggregate statistics across tagged patients); (v) structured interoperability (e.g., FHIR resources for observations) with careful scoping; and (vi) exploratory cross-walking studies to relate NLGHI trajectories to CFS strata, CCI or Elixhauser burden, or functional decline as measured by ADL scales, subject to research approvals [6–9].

CONCLUSION

NLGHI v1.0 provides a transparent, local-first approach to capturing and tracking multi-domain geriatric impairment in community and primary care settings. Its strength lies in structured repetition—the same 27-domain form, the same DSAV computation, the same normalized GHI—enabling meaningful within-patient trend interpretation without complex analytics. The software's reliance on mature scientific libraries and standardized formats aligns with reproducibility and long-term maintainability. Although not a substitute for validated frailty or comorbidity instruments, NLGHI can operate alongside them to support care planning, communication, and audit-ready documentation.

REFERENCES

1. Nightingale G, Burhenn PS, Puts M, Stolz-Baskett P, Haase KR, Sattar S, et al.

Integrating nurses and allied health professionals in the care of older adults with cancer: a report from the International Society of Geriatric Oncology Nursing and Allied Health Interest Group. *J Geriatr Oncol*. 2020;11(2):187-190. doi:10.1016/j.jgo.2019.06.012.

2. Harris CR, Millman KJ, van der Walt SJ, Gommers R, Virtanen P, Cournapeau D, et al. Array programming with NumPy. *Nature*. 2020;585:357-362. doi:10.1038/s41586-020-2649-2.
3. Virtanen P, Gommers R, Oliphant TE, Haberland M, Reddy T, Cournapeau D, et al. SciPy 1.0: fundamental algorithms for scientific computing in Python. *Nat Methods*. 2020;17(3):261-272. doi:10.1038/s41592-019-0686-2.
4. Hunter JD. Matplotlib: a 2D graphics environment. *Comput Sci Eng*. 2007;9(3):90-95. doi:10.1109/MCSE.2007.55.
5. Bray T. The JavaScript object notation (JSON) data interchange format. RFC 8259. RFC Editor; 2017. doi:10.17487/RFC8259.
6. Rockwood K, Song X, MacKnight C, Bergman H, Hogan DB, McDowell I, et al. A global clinical measure of fitness and frailty in elderly people. *CMAJ*. 2005;173(5):489-495. doi:10.1503/cmaj.050051.
7. Charlson ME, Pompei P, Ales KL, MacKenzie CR. A new method of classifying prognostic comorbidity in longitudinal studies: development and validation. *J Chronic Dis*. 1987;40(5):373-383. doi:10.1016/0021-9681(87)90171-8.
8. Elixhauser A, Steiner C, Harris DR, Coffey RM. Comorbidity measures for use with administrative data. *Med Care*. 1998;36(1):8-27. doi:10.1097/00005650-199801000-00004.
9. Katz S, Ford AB, Moskowitz RW, Jackson BA, Jaffe MW. Studies of illness in the aged. The Index of ADL: a standardized measure of biological and psychosocial function. *JAMA*. 1963;185(12):914-919. doi:10.1001/jama.1963.03060120024016.