MethodsX 14 (2025) 103155

Contents lists available at ScienceDirect

MethodsX

journal homepage: www.elsevier.com/locate/methodsx

Development of a risk assessment software for cumulative effect ☆

Effah Kwabena Antwi[1,*], Gifty Osei[2], Wiafe Owusu-Banahene[2], John Boakye-Danquah[3], Prisca-Maria Okafor[2], Kobina Korankye[2], Akua Nyamekye Darko[1], Priscilla Toloo Yohuno (Apronti)[1]

Natural Resources Canada, University of Ghana, University of Saskatchewan, Canada

ARTICLE INFO

Method name:
Risk Assessment Software for Cumulative Effects (RASCE)

Keywords:
Software
Framework
Risk assessment
Cumulative effects
Impact prioritization

ABSTRACT

Regional Risk Assessment is essential for evaluating the environmental impacts of large-scale resource development projects. However, existing Regional Asesessment (RA) frameworks often lack generalizability which hinders result standardization. To address these challenges, the Risk Assessment Framework for Cumulative Effects (RAFCE) was developed to provide a standardized approach for impact identification, prioritization, and mitigation during RA. Despite these strengths, the RAFCE's reliance on spreadsheet-based manual data entry and calculations, coupled with the absence of collaborative features, increases the risks of human error and inflates operational costs including time taken to complete an RA.

This paper proposes a software implementation of RAFCE to enhance efficiency and accuracy in the RA process. This is a novel approach that provides a platform unique of its kind for systematically evaluating the cumulative effects of resource exploration by multiple stakeholders. The development process involved three main steps:

- Developing a NoSQL Database to efficiently store and retrieves RA data,
- Implementing an API and Backend with Java Spring Boot automates critical functionalities and
- Building a React-based Frontend Development: that provides a user-friendly interface, that simplifies data entry and software interaction.

By automating calculations and improving the user interface, the proposed software mitigates the risks associated with manual processes and significantly reduces the cost and time required for the RA process, thereby enhancing its reliability and efficiency.

☆ **Related research article:** [12] E. K. Antwi et al., "Risk assessment framework for cumulative effects (RAFCE)," Frontiers in Environmental Science, vol. 10, p. 1,055,159, 2023. https://doi.org/10.3389/fenvs.2022.1055159

* Corresponding author.

E-mail addresses: effah.antwi@nrcan-rncan.gc.ca (E.K. Antwi), gbuah@ug.edu.gh (G. Osei), wowusubanahene@gmail.com (W. Owusu-Banahene), jmb357@mail.usask.ca (J. Boakye-Danquah), akua.darko@nrcan-rncan.gc.ca (A.N. Darko), priscilla.yohuno@nrcan-rncan.gc.ca (P.T. Yohuno (Apronti)).

[1] Natural Resources Canada
[2] University of Ghana
[3] School of Environment and Sustainability, University of Saskatchewan, Saskatoon

https://doi.org/10.1016/j.mex.2025.103155
Received 25 October 2024; Accepted 3 January 2025
Available online 3 January 2025

Specifications table

This table provides general information on your method.

Subject area:	Environmental Science
More specific subject area:	*Regional Risk Assessment*
Name of your method:	*Risk Assessment Software for Cumulative Effects (RASCE)*
Name and reference of original method:	*Antwi EK, Boakye-Danquah J, Owusu-Banahene W, Dabros A, Eddy IMS, Silver DA, Abolina E, Eddy Band Winder RS (2023), Risk assessment framework for cumulative effects (RAFCE). Front. Environ. Sci. 10:1,055,159. doi:0.3389/fenvs.2022.1055159*
Resource availability:	*Documentation:* GitHub repository

Background

Regional Assessment (RA) is crucial for understanding the environmental impacts of major resource development projects on a regional scale [1]. The practicality and usefulness of RA have been demonstrated and validated in several works [2–6]. including groundwater nitrate pollution [7], geological activities, and transportation projects [8,9]. These frameworks employ diverse methodologies ranging from mathematical modeling to advanced machine learning [10,11]. However, the diversity in these approaches [12], pose difficulties in comparing and standardizing RA results [13]. Additionally, methods for prioritizing impacts identified in risk assessments often rely on complex models implemented through manual error-prone spreadsheets. To address these issues, there is a need for standardized, digitized RA frameworks that promote accuracy in impact prioritization and ease of use for all stakeholders. Such frameworks should also support efficient collaboration among geographically dispersed stakeholders and lead to reduced project assessment time and costs.

Several RA software solutions have been developed to facilitate regional risk assessment. For example, Sedan et al. [14] developed Armagedom for seismic risk assessment featuring models for generating earthquakes, and vulnerability and damage estimation modules. However, this solution is limited to seismic scenarios and does not support collaborative features. Agostini et al. [15] developed a Spatial decision support system for regional risk assessment of degraded land (SYRIADE DSS), a tool for comparing and ranking potential risks associated with different contaminated sites in a region. This enables efficient distribution of limited resources for risk management leading to more sustainable environment protection practices.

Projects requiring regional risk assessments often involve complex interactions among multiple stressors over extended periods, making cumulative risk assessment (CRA) frameworks more suitable than traditional methods [16,17]. However, these frameworks are complex compared to single-stressor frameworks. Researchers have used frameworks to address different components of a regional risk assessment process. For example, frameworks for scenario modeling such as ALCES Online Scenario Analysis have been used to assess thresholds that support future planning by stimulating impacts of resource exploration on Valued Ecosystem Components (VECs) [18]. Other approaches to RA have been to collect data about past social, cultural, and environmental effects for retrospective evaluation of the impact of projects on specific VECs [19]. Community engagement approaches have been used by scientists to exclusively engage key community representatives on ways to transform cumulative effect science into policy [13]. However, the use of different frameworks and methodologies leads to inconsistencies that make it difficult to compare results [20]. The Risk Assessment Framework for Cumulative Effects (RAFCE) developed by Antwi et al. [13] addresses the above concern. RAFCE is a risk and impacts-based cumulative effects assessment framework for scoping regional cumulative effects issues. It addresses all the components of a regional risk assessment in one framework. The framework identifies key valued ecosystem components (VECs) and uses an intuitive, customized, impact prioritization model to quantify and prioritize identified impacts. In addition, due to the simplicity and robustness of the RAFCE, it is well-suited for development into a software tool. This work aims to develop a web-based, collaborative software tool that digitizes the RAFCE. The software must be able to aggregate all Valued Ecosystem Components (VECs), provide inputs on preventive and mitigation recommendations for identified threats and consequences from the inputs of experts, and develop an impact prioritization model. In this research, we aim to:

1. Design and develop user-friendly, robust, and reliable Risk Assessment Software for Cumulative Effect (RASCE) that can perform comprehensive impact prioritization as described in the framework [13].
2. Generate intelligible visual reports automatically, containing project-based risk assessment and the results of the impact prioritization model.
3. Implement collaborative features that enable seamless collaboration among geographically dispersed stakeholders

The proposed software significantly enhances the efficiency and accuracy of regional risk assessments, thereby supporting more informed and effective decision-making for major resource development projects.

Methodology

We developed a cumulative risk assessment software application using the client-server software architecture [21]. The software development was carried out in three parts: the database model development, the Application programming interface (API) development, and the frontend development. These represent the three main parts of the software as indicated in Fig. 1. Users will interact with the application through a web-based interface that communicates with the backend via an API through the Hypertext Transfer Protocol (HTTP). The backend uses a Model View Controller (MVC) architecture variant and communicates with a No-SQL database.

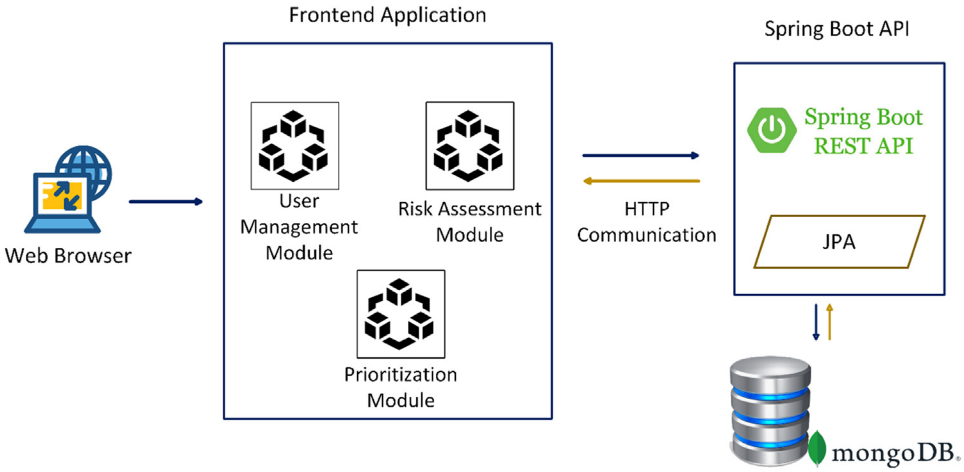

Fig. 1. Software system architecture.

A. Database Development

A NoSQL database was selected to accommodate the flexibility and varied structure of data generated throughout the risk assessment and prioritization process. We prioritized the need for flexibility in storing and managing data, as different projects may specify varying criteria and scores for impact prioritization. Given this dynamic nature, we opted for a NoSQL database, specifically MongoDB, instead of a traditional relational SQL alternative such as MySQL. MongoDB stores data in binary representation known as BSON (Binary JSON). Each document is a key-value pair structure that can hold nested fields, arrays, and other complex data types. Unlike relational databases with rigid schemas, MongoDB documents in the same collection can have varying structures and hence support evolving data models without costly migrations. MongoDB was also chosen due to its efficient querying and indexing capabilities [22], and seamless integration with our API development framework, Spring Boot.

The database was designed around two primary entities: "profile" and "project". The “profile” entity captures all user-related activities, including authentication and authorization processes using JSON Web Tokens (JWT). It ensures secure access control and user management within the system. The “project” entity encompasses all activities associated with regional risk assessment projects including data related to stressors, impacts, preventive and mitigating factors and scenarios, and the prioritization model. Each document that describes these entities is structured to facilitate efficient storage and retrieval of information, enabling real-time data sharing and collaboration among geographically dispersed stakeholders. The complete database model is shown in Fig. 2.

B. *Application Programming Interface (API) Development*

The API was developed with Java Spring Boot, following the standard model-controller repository-service format offered by Spring Boot [23]. A “*profile*” defines all entry points for actors to interact with other services in the API. We identified three main actors for the software system:

1. A principal investigator (PI), who creates new risk assessment projects and manages other stakeholders invited to collaborate on the project.
2. A system administrator (Administrator) of the software who manages all users and has the clearance to revoke PIs and projects and assist with software management.
3. A stakeholder (Collaborator) who collaborates on a project and is invited by the PI or applies to join a project and is approved by the PI.

The use case diagram in Fig. 3 shows how each actor interacts with the software. Through this interface, actors can collaborate, create risk assessment diagrams that specify stressors, mitigating factors, and impacts, and invite new collaborators. An important service we defined is the *Diagram* service. This service contains the main business logic of the RAFCE software and implements all functions from the specification of RA nodes to prioritization. Additionally, collaborative functions that allow users to create and edit nodes in real time are specified here.

Furthermore, we leveraged MongoDB’s “*Change Streams*” [24] to get real-time Create, Read, Update, and Delete (CRUD) updates on the required data collections and implemented server-sent events to push the formatted updates to the client-side application. Reactive programming paradigms such as the Flux and Mono datatypes were used to ensure all events triggered during the diagram lifecycle were captured. The software defines the following nodes:

1. Risk Source: An activity or project (s) embedded in it a source of potential harm or a situation with the potential to cause harm.
2. Critical Event (CE): the main occurrence that causes harm, loss, or damage.

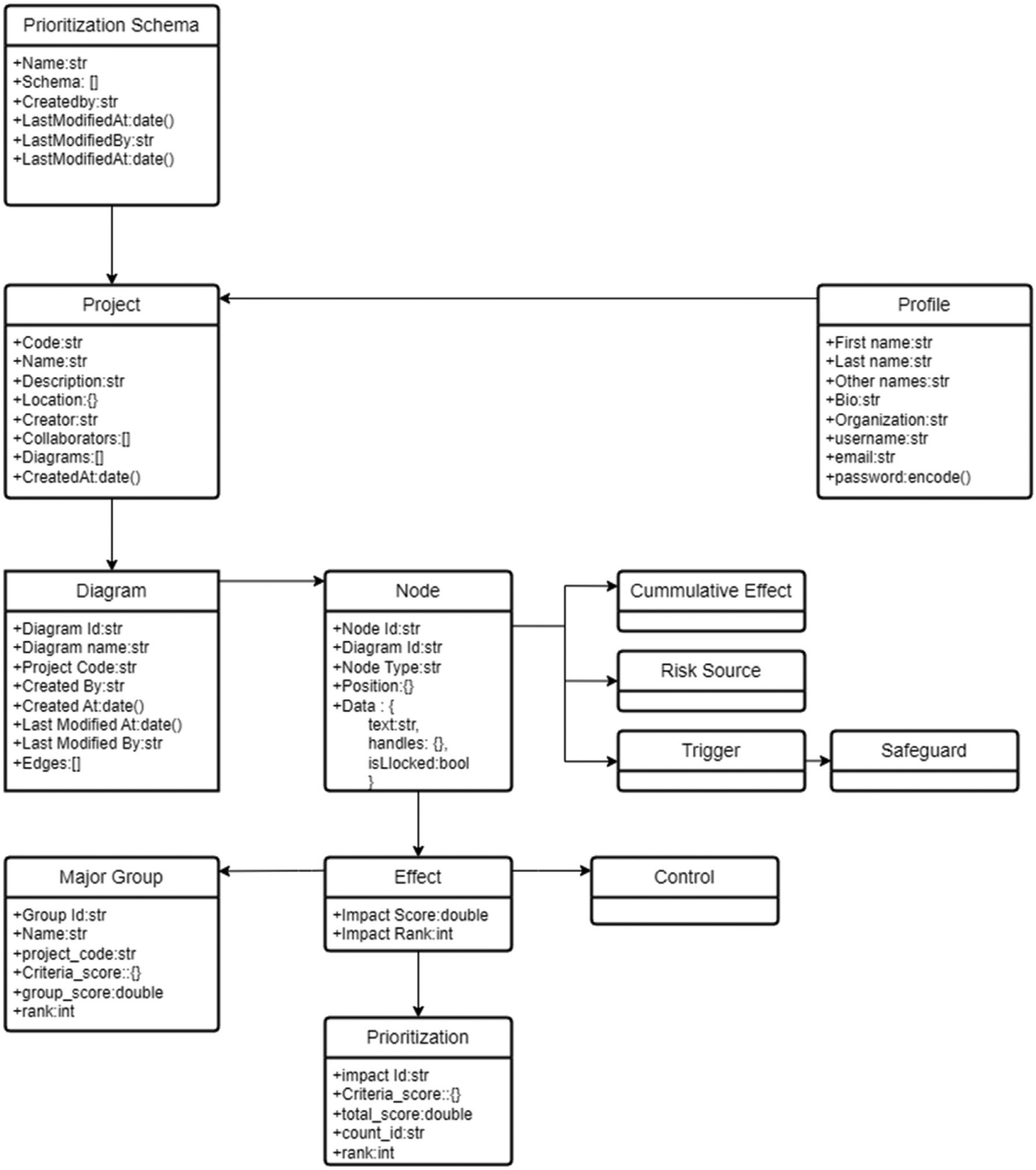

Fig. 2. NoSQL database design.

3. Triggers: situations that lead to a critical event.
4. Effects: repercussions of the occurrence of a critical event.
5. Safeguards: measures to prevent triggers from causing a CE.
6. Controls: measures to reduce the severity of effects

A *Prioritization* service was also implemented to define the functionality of the prioritization model specified in [13]. This functionality analyses a project's impacts and scores them according to Eq. (1).

$$T_i = S_i + U_i + C_i + \sum_{1}^{j} F_j \tag{1}$$

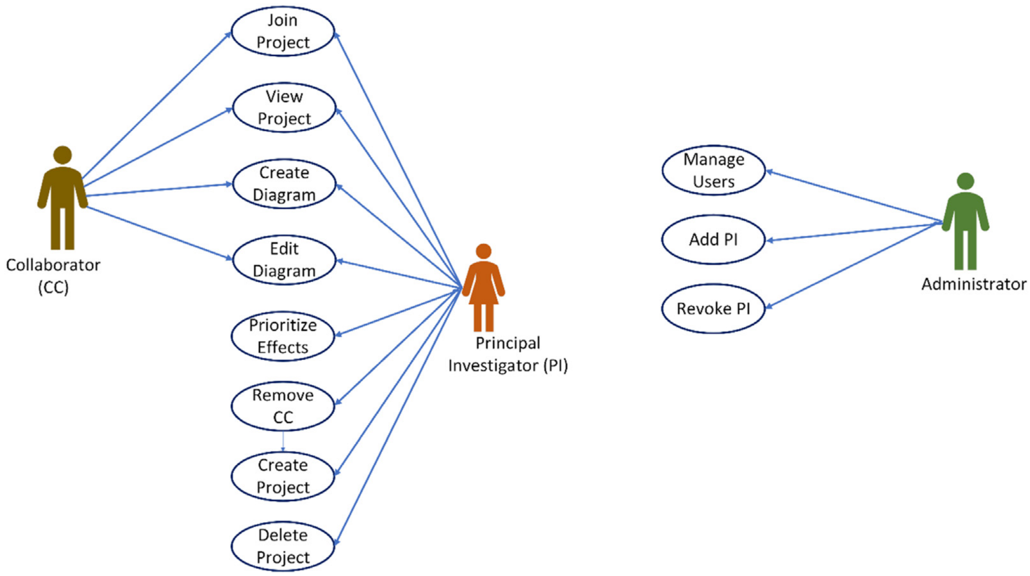

Fig. 3. Use case diagram for main actors.

Where S_i represents the key stakeholder interest, U_i captures the underlying issues and C_i denotes the frequency occurrence of Effects throughout the project. F_j represents all other factors considered for the prioritization of Effects such as recoverability, severity of impact, geographic extent, etc. as specified in the initial framework. A "count groups" property was created to keep track of the number of occurrences of each Effect to calculate C_i. PIs and Collaborators select impacts they reckoned to be the same. A "total count" variable represents the number of items in the count group. C_i was then computed as a fraction of the Effect's total count to the maximum total count for every Effect in the project.

After the scores (Ti) were computed, they were ranked in descending order. Each Effect was also placed in a major category, and all major categories were ranked in descending order. An "impact groups" property was created to track and rank the major categories. The category score was computed as the sum of all the individual impact scores in the group. Hence the rank of a major category depended on the cumulative ranks of the individual Effects in the category.

C. Frontend Development

The primary framework used in developing the frontend was React, complemented by several auxiliary libraries to handle specific tasks. React Router Dom (v6.14.1) was employed for in-app navigation, facilitating efficient route management. React Toastify (v9.1.3) was integrated to provide in-app notifications and React Tooltip (v5.20.0) was used to display helpful messages upon hovering over UI components. React Flow (v11.8.2) was used to implement the risk assessment drag-and-drop functionality, and Xlsx (v0.18.5) was incorporated to convert HTML tables into Excel format after prioritization. The Mapping functionalities and the displaying of the geographic locations of projects were powered by Leaflet (v1.9.4) and React Leaflet (v4.2.1), enabling dynamic map creation and management. Interactions between the frontend and the API were efficiently managed using Axios (v1.4.0), which handled all data fetching and submission operations. The frontend application consisted of 3 main modules: the user management module, the risk assessment module, and the prioritization module as shown in the architecture diagram in Fig. 1. A high-level flow of actions performed by actors at the frontend is depicted in Fig. 4. The user management module authenticates users via a username and password, checks each user's role, and gives access to the risk assessment and prioritization modules. The risk assessment module allows the specification of risks, critical events, triggers, safeguard effects, and controls for valued ecosystem components. The effects identified in the RA process are then prioritized and ranked by the prioritization module. The detailed process of using the RA and prioritization modules is shown in Fig. 5.

D. Deployment Strategy

The frontend of our application was hosted on Vercel, while the backend was hosted on Render. We integrated our GitHub repository with these services to enable continuous deployment. This integration ensures that updates are automatically deployed upon each Git push, making new changes live immediately after merging into the main branch. This approach streamlines the deployment process, allowing for rapid updates and iterations, and ensures high availability and immediate responsiveness to application requirements or fixes. Different environments (development, staging, production) were managed using separate configuration files and environment variables to ensure security and scalability.

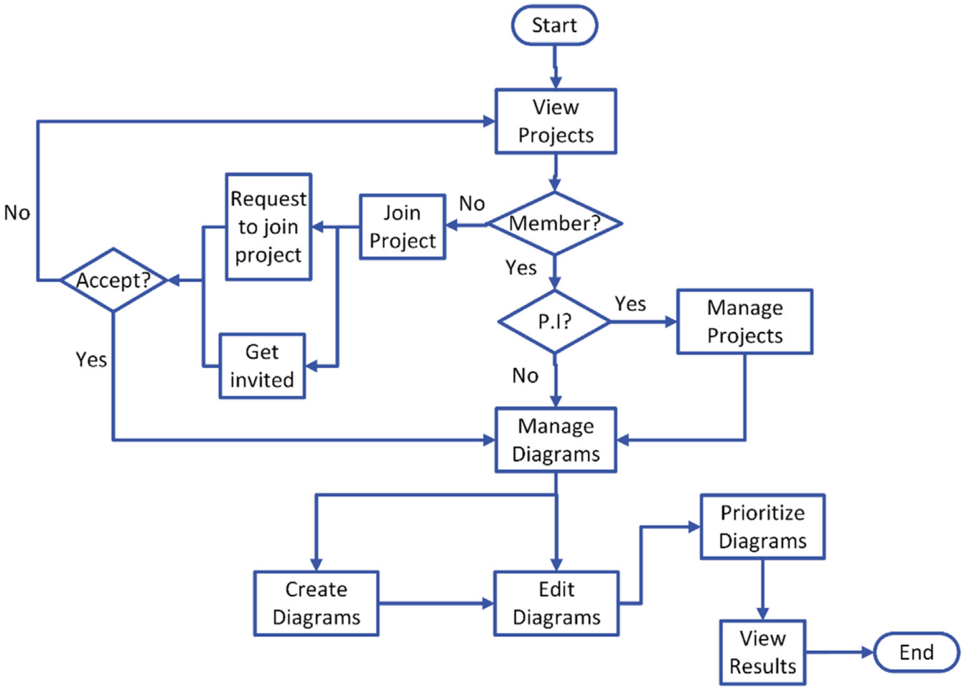

Fig. 4. High-level flow of actor actions from login to prioritization.

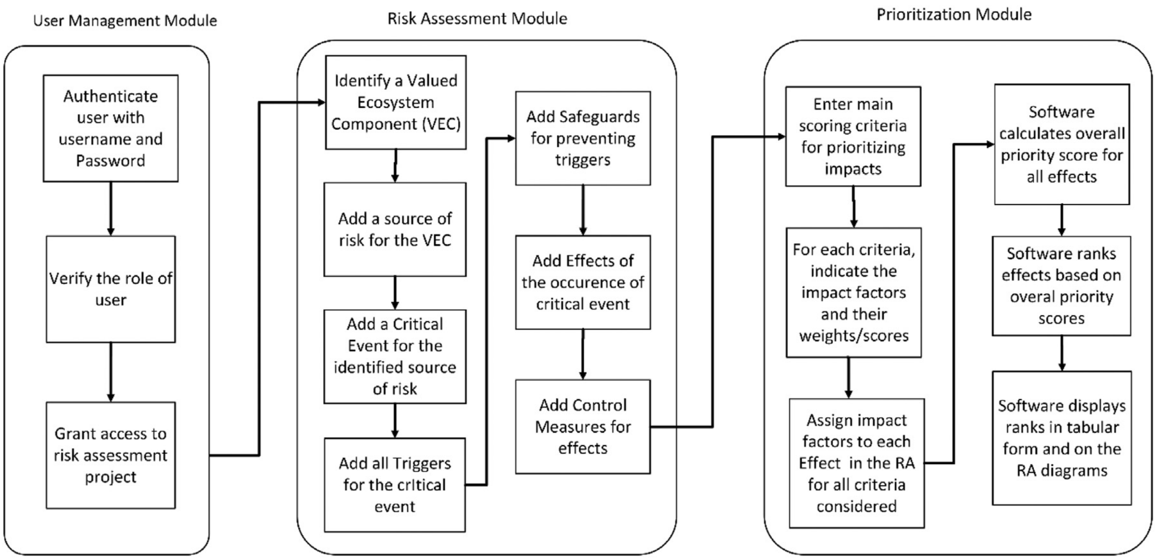

Fig. 5. Detailed module interactions.

Method validation

The developed software application aimed to facilitate regional risk assessments through a web-based, collaborative platform. The application integrated RAFCE to support identifying, prioritizing, and managing of environmental impacts. Upon logging into the application, users are presented with a dashboard with an overview of project metrics, recent activities, and notifications. A mapping feature on the dashboard allows easy viewing and selection of projects according to geographical area as shown in Fig. 6.

Users navigate to the RA project interface using the “work area” menu from the side navigation bar. Since projects are regionally based, selecting a project’s location from the map intuitively takes you to the work area of the RA project. The work area of each RA project gives a brief description of the selected project and lists all risk assessments started or completed for each risk source as

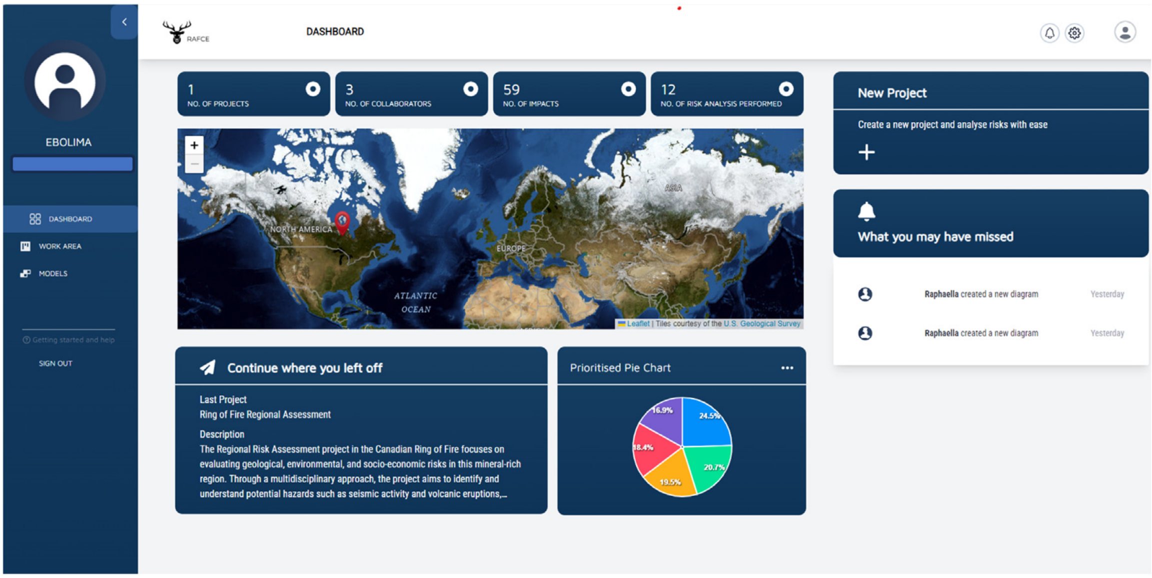

Fig. 6. RAFCE software dashboard.

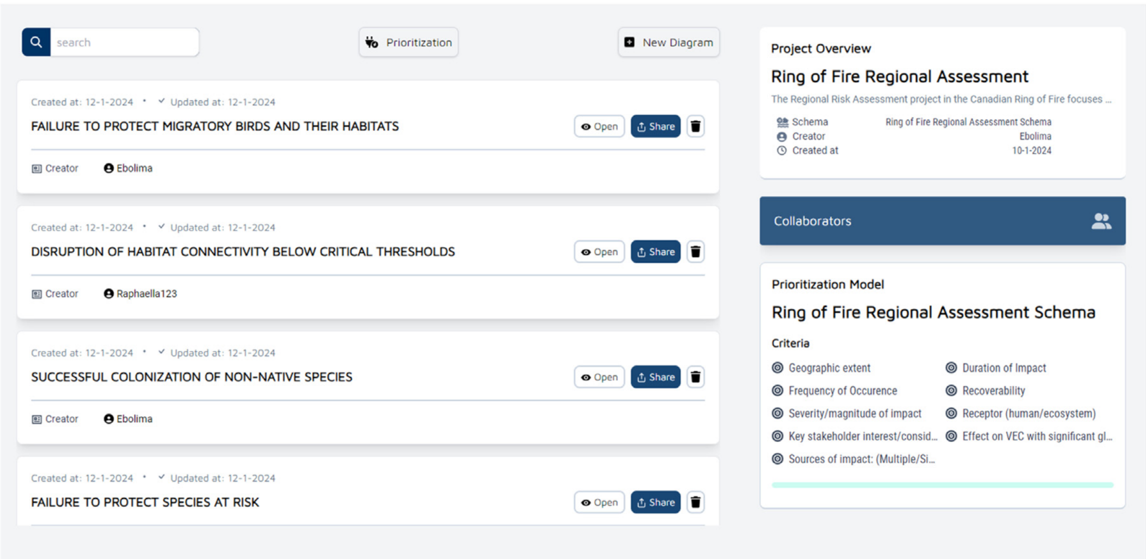

Fig. 7. Work area of select projects listing critical events.

shown in Fig. 7. This interface also shows, at the right-hand corner the various criteria for scoring effects. The criteria can be edited, or a new set of criteria defined for different projects.

Selecting a risk source from the list leads to the RA diagram interface where critical events, triggers, effects, safeguards, and controls can be modified. Prioritization criteria and corresponding values are defined in the model section and assigned to each effect before prioritization starts , as shown in Fig. 8. Figure 8 shows the full view of the RA diagram interface and the zoomed-in view of the effects and trigger interfaces.

On the prioritization interface, all effects are displayed and color-coded to identify different critical events as shown in Fig. 9. Effects are counted for duplicates and grouped into major categories. Clicking the Prioritize button then calculates priorities for all effects and ranks them. The result of the prioritization is shown as rank values on effect nodes and can also be viewed as a table and downloaded as an Excel file.

Additionally, the output of the prioritization is displayed as numbered ranks on the top right corner of all effects and can also be viewed in tabular form as shown in Fig. 10. The table can also be downloaded as an Excel file.

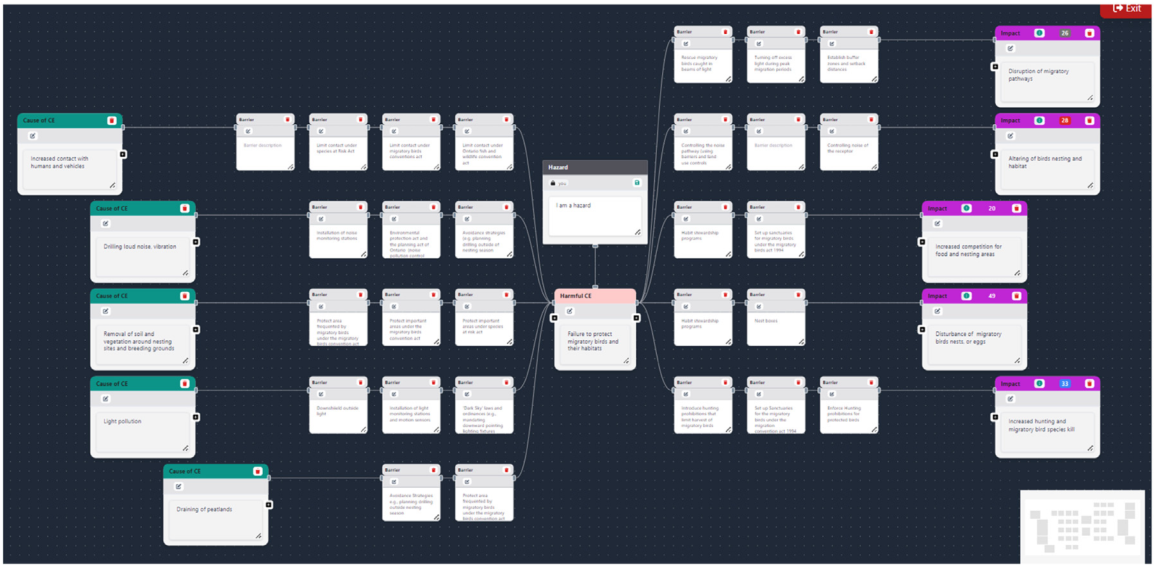

Fig. 8. Risk analysis diagram interface (informed by the RAFCE and the ISO 31010 Bow-tie Risk Assessment Tool (BRAT).

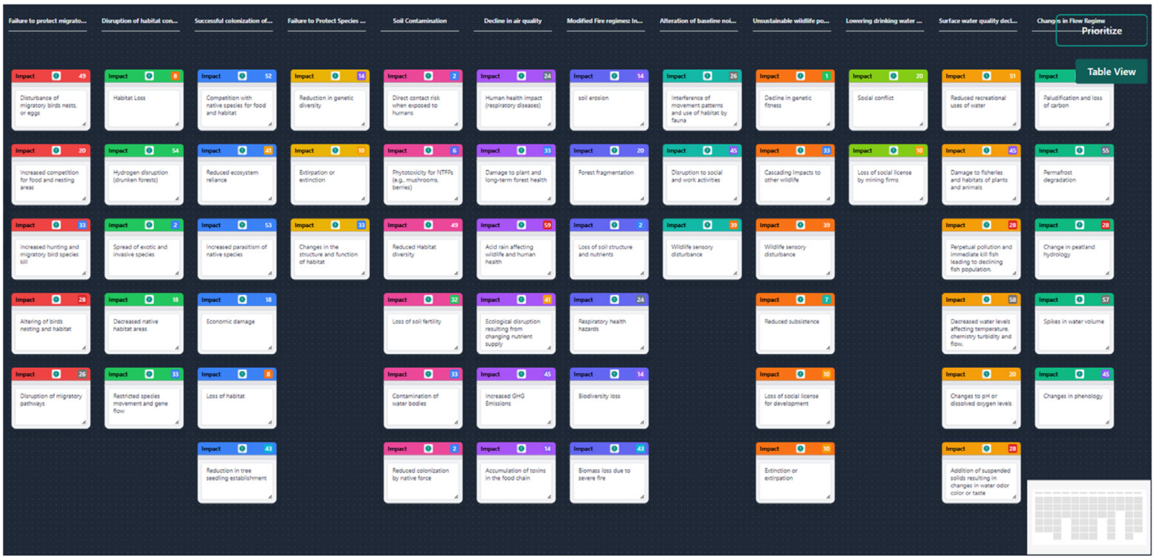

Fig. 9. Prioritization interface.

Finally, the software has been tested and reviewed in multiple scenarios and across various forums where relevant feedback was provided and subsequently incorporated into the design and architecture of software. For example, the software design and content were informed by feedback from users involving scientists, science managers, and science policy experts at the Environmental Protection Agency of Ghana, the Canadian Wildlife Service, Environment and Climate Change Canada, Health Canada's Social Determinants of Health Working Group, and by the Impact Assessment Agency of Canada. Feedback received from these engagements highlighted the software's ability to include Indigenous knowledge throughout the resource development lifecycle. Furthermore, the users expressed appreciation for the software's capability to assess the risks and impacts of both past and ongoing resource development project. This positive feedback informed the decision to operationalize the software in regions that had completed resource exploration projects such as the Alberta Oil Sands Region and ongoing mining projects in the Apitipi Anicinapek Nation. Other reviewers provided guidance on how to improve the multiuser functionalities of the software and to refrain from the use of sensitive words such as "threats" to more inclusive phrases such as "cause of harmful cumulative effects". In addition, scientists within the Cumulative Effects Hub Program at Natural Resources Canada reviewed the software to ensure that the tool aligns with all the necessary Information Technology security requirements. Finally, the Forest Information Service Division within the Canadian Forest Service

Major Impact Group	Group Count	Impacts	Count	Geographic Extent	Duration Of Impact
Food sovereignty and security (access quality and quantity)humans	3	Phytotoxicity for NTFPs (e.g., mushrooms, berries)	1	1	1
		Accumulation of toxins in the food chain	1	0.33	1
		Reduced subsistence	1	0.33	1
Destruction of fish population	1	Perpetual pollution and immediate kill fish leading to declining fish population.	1	0.33	0.67
Decline in human health	7	Economic damage	1	1	1
		Direct contact risk when exposed to humans	1	1	1
		Human health impact (respiratory diseases)	2	0.33	0.67
		Acid rain affecting wildlife and human health	1	0.33	0.67
		Disruption to social and work activities	1	0.33	1
		Reduced recreational uses of water	1	0.67	1
Heightened tension/dispute with communities	3	Loss of social license for development	2	0.33	0.67
		Social conflict	1	1	0.67
Climate variability	1	Increased GHG Emissions	1	0.33	1
		Reduction in tree seedling establishment	1	0.67	1
		Reduction in genetic diversity	1	0.67	1

Fig. 10. Tabular view of prioritization results.

provided feedback on the outlined government legislation stored in the software's database and provided the necessary guidance to align this to meet all ethical guidelines.

Limitations and future work

While the RAFCE software significantly improves the efficiency and accuracy of Regional Risk Assessments, limitations still exist. The software relies on manual data entry from experts, which can introduce errors in the automated calculations if the input data entered is incorrect or inconsistent. Additionally, the software's collaborative features may place a significant load on the backend and database during intensive collaborative sessions, potentially leading to performance bottlenecks. In future work, we intend to include integration with existing risk assessment systems through data import/export functionality as well as plugins through API-based approaches. With this functionality, users can export risk assessment data from other tools in standard formats like JSON or CSV and import it into RAFCE software to leverage its custom prioritization framework. Additionally, we envision providing RESTful APIs allowing external systems to programmatically access the prioritization functionality by sending risk assessment data and receiving prioritized results. These potential integrations would enhance interoperability and streamline workflows for users.

Ethics statements

Not Applicable.

Credit Author Statement

Effah Kwabena Antwi: Software Validity tests, Investigation, Conceptualization, Methodology, Data curation, Visualization, Supervision, Writing- Original draft preparation, Writing- Reviewing and Editing. **Gifty Osei:** Software, Software Validity tests, Visualization, Investigation, Data curation, Writing- Original draft preparation, Supervision, Writing- Reviewing and Editing. **John Boakye Danquah:** Conceptualization, Methodology, Data curation, Reviewing and Editing. **Wiafe Owusu-Banahene:** Software, Software Validity tests, Supervision. **Prisca-Maria Okafor:** Visualization, Investigation, Software, Validation. **Kobina Korankye:** Visualization, Investigation, Software, Validation. **Akua Nyamekye Darko:** Investigation, Software Validity tests, Writing- Original draft preparation, Writing- Reviewing and Editing. **Priscilla Toloo Yohuno (Apronti):** Investigation, Software Validity tests, Writing- Reviewing and Editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

The software development is supported by the Natural Resources Canada,-Canadian Forest Service Cumulative Effects Program.

References

[1] J. Blakley, B. Noble, K. Vella, J. Marty, K. Nwanekezie, K. Fedoroff, Lessons learned, best practices and critical gaps in regional environmental assessment: a synthesis of Canadian and international literature, Prepared Social Sci. Humanities Res. Council Canada Impact Assess. Agency Canada 28 (2020) 2020 *Accessed October*.
[2] D. Johnson, et al., Improving cumulative effects assessment in Alberta: regional strategic assessment, Environ. Impact. Assess. Rev. 31 (5) (2011) 481–483.
[3] W.D. Walter, et al., Regional assessment on influence of landscape configuration and connectivity on range size of white-tailed deer, Landsc. Ecol. 24 (2009) 1405–1420.
[4] M.P. Ledru, M.L.F. Salatino, G. Ceccantini, A. Salatino, F. Pinheiro, J.C. Pintaud, Regional assessment of the impact of climatic change on the distribution of a tropical conifer in the lowlands of South America, Diversity Distributions 13 (6) (2007) 761–771.
[5] L.M. Arnold, K. Hanna, B. Noble, Freshwater cumulative effects and environmental assessment in the Mackenzie Valley, Northwest Territories: challenges and decision-maker needs, Impact Assess. Project Appraisal 37 (6) (2019) 516–525.
[6] D. Kirchhoff, D. McCarthy, D. Crandall, G. Whitelaw, Strategic environmental assessment and regional infrastructure planning: the case of York Region, Ontario, Canada, Impact Assess. Project Appraisal 29 (1) (2011) 11–26.
[7] Y. Teng, R. Zuo, Y. Xiong, J. Wu, Y. Zhai, J. Su, Risk assessment framework for nitrate contamination in groundwater for regional management, Sci. Total Environ. 697 (2019) 134102.
[8] J. Alberti, Modern tools for holistic assessment of ambiguity and risk: a case study of three Latin American mega transport projects, J. Mega Infrastruct. Sustain. Develop. 2 (1) (2020) 87–125.
[9] J. Clarke, E. Obrien, A multi-hazard risk assessment methodology, stress test framework and decision support tool for transport infrastructure networks, Transport. Res. Procedia 14 (2016) 1355–1363.
[10] N. Chen, L. Chen, Y. Ma, A. Chen, Regional disaster risk assessment of China based on self-organizing map: clustering, visualization and ranking, Int. J. Disaster Risk Reduct. 33 (2019) 196–206.
[11] E.K. Antwi, W. Owusu-Banahene, J. Boakye-Danquah, B. Asare, A.F.B. Frimpong, An automated method for developing search strategies for systematic review using Natural Language Processing (NLP), MethodsX 10 (2023) 101935. https://www.sciencedirect.com/science/article/pii/S2215016122003120.
[12] E.E. Hodgson, B.S. Halpern, T.E. Essington, Moving beyond silos in cumulative effects assessment, Front. Ecol. Evol. (2019) 211.
[13] E.K. Antwi, et al., Risk assessment framework for cumulative effects (RAFCE), Front. Environ. Sci. 10 (2023) 1055159.
[14] O. Sedan, C. Negulescu, M. Terrier, A. Roulle, T. Winter, D. Bertil, Armagedom—A tool for seismic risk assessment illustrated with applications, J. Earthquake Eng. 17 (2) (2013) 253–281.
[15] P. Agostini, L. Pizzol, A. Critto, M. D'Alessandro, A. Zabeo, A. Marcomini, Regional risk assessment for contaminated sites Part 3: spatial decision support system, Environ. Int. 48 (2012) 121–132.
[16] S.S. Gallagher, G.E. Rice, L.J. Scarano, L.K. Teuschler, G. Bollweg, L. Martin, Cumulative risk assessment lessons learned: a review of case studies and issue papers, Chemosphere 120 (2015) 697–705.
[17] A.S. Lewis, S.N. Sax, S.C. Wason, S.L. Campleman, Non-chemical stressors and cumulative risk assessment: an overview of current initiatives and potential air pollutant interactions, Int. J. Environ. Res. Public Health 8 (6) (2011) 2020–2073.
[18] Sullivan, M.G. (2009). Cumulative effects assessment of the North Saskatchewan River watershed using ALCES. Available at: http://nswa.ab.ca/cumulative_effects.
[19] Manitoba Government and Manitoba Hydro (2015). Regional cumulative effects assessment for hydroelectric developments on the Churchill, Burntwood and Nelson River systems phase II report, Part II hydroelectric project description. Avaliable At: https://www.hydro.mb.ca/docs/regulatory_affairs/pdf/rcea/rcea_phase2_part_ii_hydroelectric_development_project_description.pdf.
[20] E.E. Hodgson, B.S. Halpern, T.E. Essington, Moving beyond silos in cumulative effects assessment, Front. Ecol. Evol. 7 (2019) 211, doi:10.3389/fevo.2019.00211.
[21] M.D. Hanson, The client/server architecture, in: Server Management, Auerbach Publications, 2000, pp. 17–28.
[22] R. Chopade, V. Pachghare, MongoDB indexing for performance improvement, in: ICT Systems and Sustainability: Proceedings of ICT4SD 2019, 1, Springer, 2020, pp. 529–539.
[23] C. Walls, *Spring in Action*. Simon and Schuster, 2022.
[24] I. MongoDB. "Change streams." https://www.mongodb.com/docs/manual/changeStreams/ (accessed October 18, 2024, 2024).