

Article

# Collaboration System for Multidisciplinary Research with Essential Data Analysis Toolkit Built-In

Laura I. Garay-Jiménez <sup>1,\*</sup> , Jose Fausto Romero-Lujambio <sup>2</sup>, Amaury Santiago-Horta <sup>2</sup> , Blanca Tovar-Corona <sup>1</sup>, Pilar Gómez-Miranda <sup>3</sup>  and Miguel Félix Mata-Rivera <sup>4</sup> 

<sup>1</sup> Instituto Politécnico Nacional-UPIITA, SEPI, LIPS, CDMX, Mexico City C.P. 07340, Mexico; bltovar@ipn.mx

<sup>2</sup> Instituto Politécnico Nacional-UPIITA, CDMX, Mexico City C.P. 07340, Mexico; jromerol0902@alumno.ipn.mx (J.F.R.-L.); asantiagoh2300@alumno.ipn.mx (A.S.-H.)

<sup>3</sup> Instituto Politécnico Nacional-UPIITA, SEPI, CDMX, Mexico City C.P. 08400, Mexico; pgomezmi@ipn.mx

<sup>4</sup> Instituto Politécnico Nacional-UPIITA, SEPI, Laboratory of Geo-Spatial Intelligence and Mobile Computing, CDMX, Mexico City C.P. 07340, Mexico; mmatar@ipn.mx

\* Correspondence: lgaray@ipn.mx; Tel.: +52-5557296000 (ext. 56845)

**Abstract:** Environmental research calls for a multidisciplinary approach, where highly specialized research teams collaborate in data analysis. Nevertheless, managing the data lifecycle and research artifacts becomes challenging because the project teams require techniques and tools tailored to their study fields. Another pain point is the unavailability of essential analysis and data representation formats for querying and interpreting the shared results. In addition, managing progress reports across the teams is demanding because they manage different platforms and systems. These concerns discourage the knowledge-sharing process and lead to researchers' low adherence to the system. A hybrid methodology based on Design Thinking and an Agile approach enables us to understand and attend to the research process needs. As a result, a microservices-based architecture of the system, which can be deployed in cloud, hybrid, or standalone environments and adapt the computing resources according to the actual requirements with an access control system based on users and roles, enables the security and confidentiality, allowing the team's lead to share or revoke access. Additionally, intelligent assistance is available for document searches and dataset analyses. A multidisciplinary researchers' team that uses this system as a knowledge-sharing workspace reported an 83% acceptance.

**Keywords:** data visualization; multidimensional dataset; fuzzy searching; collaborative research; t-SNE; microservices; approximate string-matching; knowledge-sharing behaviors



**Citation:** Garay-Jiménez, L.I.; Romero-Lujambio, J.F.; Santiago-Horta, A.; Tovar-Corona, B.; Gómez-Miranda, P.; Mata-Rivera, M.F. Collaboration System for Multidisciplinary Research with Essential Data Analysis Toolkit Built-In. *Information* **2023**, *14*, 626. <https://doi.org/10.3390/info14120626>

Academic Editors: Valentina Casola and Dimitris Apostolou

Received: 12 September 2023

Revised: 11 November 2023

Accepted: 20 November 2023

Published: 21 November 2023



**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Nowadays, to solve complex problems, multidisciplinary research groups must share datasets, preliminary results, and background information to later process, report, and analyze information and documents. The collaboration allows the research team to analyze integrated results and monitor products, providing continuity to the investigation and guiding the decision-making processes while maintaining control over the information's flow.

On the one hand, the World Wide Web (WWW) has promoted globalization and outsourcing, reducing the product development cycle time. In addition, various new approaches and tools were introduced and applied in the sharing of information methods [1].

However, the research activity has become a complex interaction where knowledge-sharing behavior (KSB) is required, not just information sharing, to increase the synergic interaction [2]. Several works introduced Knowledge Sharing (KS) as one of the significant knowledge management activities. Moreover, the KSB has a vital role in every researcher and academic organization and its research goals because it is involved in the innovation and synergic work in this highly specialized community. Akhavan et al. presented

an extensive study about the factors that promote the intention (KSI) towards a KSB. They considered 40 reported factors related to intrinsic motivation and extrinsic motivation factors and the sharing methods in a research community of 317 members, with 91% male and 8% female with education level distributed into PhD (88), MA (143), BA (86) in three positions: Senior, Middle Manager, and Researchers. They found that factors such as culture, organizational structure, and use of technology are decisive in KSB, but the main factors are the intrinsic motivation factors. In the researchers' case, performing routine and daily tasks is less attractive than being involved with challenging tasks that need solving a problem or an issue. So, intrinsic and extrinsic motivations should be carefully considered [3]. In addition, in this specific case, it is crucial to consider that solving collective-action problems requires three key processes: cooperation, learning, and distribution of resources; the diversity and abundance of actors and institutions and the structural relations among them are essential properties as in an ecological system [4].

Nowadays, the availability of specialized tools required by each multidisciplinary project team may hinder the data-sharing process for intermediate transformation across different teams, partly due to the unavailability of a standard set of tools and data representation formats for querying and interpreting the research artifacts being produced. Some open-source visualization platforms such as SNIK Graph—Visualising Knowledge about Management of Hospital Information Systems are based on Node.js programming [5], Python [6], or WEKA that helps analysis and visualization of datasets, and their result could be shared [7–9]. However, they require a long learning curve about programming or user interface understanding, minimizing the acceptance or preventing their generalized use in a multidisciplinary researcher's community.

To consolidate the KS, management of progress reports across teams in the project is a way that allows quick evaluation or feedback. However, it could be a complicated task, as different teams may decide to manage their progress using various platforms and electronic systems, such as email, instant messaging, or cloud storage services.

Therefore, identifying the daily tools required to analyze and visualize multivariable datasets for Information Sharing (IS) and generate strategies to promote the sharing is an open issue to study. All these factors may affect the knowledge-sharing process across different teams and lead to researchers having a low adherence to technological platforms that promote collaborative work.

This paper aims to propose a model for improving knowledge sharing, proving a methodology and tools to assist multidisciplinary research groups in sharing information under a safe Web platform using a human-centric experience design.

In this paper, our contribution is two-fold:

- In the first step, we address the identification of the needs, pain points, and challenges directly with the final user and propose an incremental solution, usable since the first version, covering the basic needs in information management, fuzzy search of documents, and metadata, and compute essential statistic and display information of the datasets transparently.
- Secondly, we implement tools such as visual representations of n-variables, computing linear and nonlinear regressions of any numerical pair of selected variables, and displaying information in several distinct forms, with particular emphasis on t-SNE and PCA plots for supervised and non-supervised database clusters.

The rest of this paper is divided into six sections. Section 2 presents the study's background. Section 3 presents the materials and methods for the model generation, implementation, and testing of the experiments into the designed Web platform. Section 4 describes the results, and Section 5 discusses the findings with respective recommendations based on the study's objectives. Finally, Section 6 closes the paper with the conclusion.

## 2. Theoretical Background

As mentioned, Akhavan et al. presented an extensive study about the factors that promote the intention (KSI) to a behavior (KSB) in 317 participants with different education

levels and positions in a research community. They concluded that the intrinsic motivation factors are crucial in knowledge-sharing behavior. Consequently, the KS process across different teams is affected, leading researchers to low adherence to technological platforms that promote collaborative work [3,5]. So, before implementing these technological methods, design thinking (DT) is considered an approach that prioritizes understanding each team member's overall workflow and human-centric experience, enabling system designers to understand and attend to the research process needs.

### *2.1. Collaborating System Infrastructure*

Information-sharing requires data governance related to authority and control over data, and such authority matters through decision-making in data-related. Moreover, data governance should focus on data flow and the systems through which data is collected, managed, and used. The organizational approach to data governance is used in this case, emphasizing structure, responsibility, accountability, and reporting [10]. So, this approach sets up organizational structures for data governance and treats data governance as a defining authority, which is then combined with the Authorization, Authentication, and Accounting methodology (AAA) using the principle of top-level design [10,11].

Implementing a system that enables researchers in a multidisciplinary project to share information and research artifacts calls for developing a purpose-specific data management tool. Access to the proposed system is provided using a Web application, as the availability of up-to-date Web browsers supporting current technologies has become ubiquitous across all principal desktop and mobile operating systems [12,13]. Software development current practices have identified the benefits of a Microservices Architecture (MSA) in opposition to a monolithic design, as they allow for granular services that can be tightly integrated, providing benefits such as high availability, flexibility, scalability, and speed [14].

Of particular importance, the flexibility of such a software architecture allows deployment on different computing environments, such as a public or private cloud, on-premises, or even a mixed approach: different components of the application can be deployed on different computing environments without affecting the application functionality or usability. Cloud computing allows on-demand scalability of deployed services, increasing or decreasing the number of computing resources based on the requirements of each software team, and depending on the specific design of the microservice, vertical or horizontal scaling can be achieved.

### *2.2. Multidimensional Data Analysis and Visualisation*

Multidimensional data analysis and visualization are steadily evolving in data science. Trends in this area are focused on interactive visualization that allows users to explore data in multiple dimensions efficiently, using dimensionality reduction techniques such as Principal Component Analysis (PCA) and tDistributed Stochastic Neighbor Embedding (t-SNE) to transform data into lower dimensional spaces. In addition, 3D visualization could represent complex relationships between variables and create specialized tools to visualize extensive multidimensional datasets [15].

Moreover, integrating machine learning (ML) techniques with visualization has led to the creation of tools that automatically identify patterns and relationships in data, simplifying the analysis process.

These multidimensional data analyses and visualization advances have been implemented in several research fields. For instance, in biology, specific visualization techniques have been developed for genomic data, allowing the representation of gene expression patterns and relationships between genes [16], or in the analysis of complex high dimensional systems associated with interaction in smart cities such as a study on road accident open data in Medellín [17]. Moreover, data visualization has been helpful in qualitative and subjective analysis, complemented with metrics that calculate the level of similarity or differentiation of Big Data model components [18].

The graphical visualization of a high-dimension system could be challenging to interpret, so transformations are required. The t-SNE method emerges as a compelling addition, emphasizing data visualization because it is a nonlinear approach. Its applicability extends to various fields with remarkable results. It has gained relevance in environmental sciences by reducing the dimensionality of environmental data. In addition, it allows for more straightforward and more understandable visualizations, conducting well-informed decision-making, and identifying critical patterns. It has been employed to visualize environmental surveillance data, such as enhancing monitoring capabilities in smart cities [19], analyzing geochemical data of groundwater [20], and improving the quality of vegetation mapping data [21]. The effectiveness of t-SNE in interpreting complex data makes it a valuable tool for environmental data research and analysis.

### 3. Materials and Methods

To generate this technological method for knowledge sharing, the Design Thinking methodology is proposed as an approach that prioritizes understanding each team member's overall workflow and the human-centric experience, enabling system designers to understand and attend to the research process needs and to identify opportunities and challenges to be covered. A hybrid model for a technical proposal of a specific platform with functionality and several common tools that promote the KSB has been defined based on a specific requirement of the final users.

#### 3.1. Hybrid Methodology

The first stage, empathy, focuses on understanding the environment and user needs and putting oneself in their place to detect the system requirements accurately. Information-gathering tools such as researchers' group meetings, personalized interviews, observation in the workplace, and surveys were used. The knowledge of the researcher's workflow, feelings about it, and personal perception of his activity were used to create the empathy map for each user type (role). Then, pain points, improvement opportunities, and challenges were identified. With all this information, the roadmap is created, and the weighted need statements are generated.

In the second stage, the problem to be solved is defined using the "Point of View" (POV) technique. Then, the design team carefully analyses the detected findings, the needs of the user, and the interests, and they are included in the road map. Then, a brainstorming technique is used to generate possible solutions out of the box (big ideas). The ideas were grouped by similarity and prioritized according to the feasibility and importance assigned by the designers. Finally, the solution with higher weighting is considered, looking for a system that provides increased functionality. These solutions are presented to the researchers, and their feedback is considered in the mission statements (hills). These statements (hills) are defined considering who does the action (who), how it will be done (how), and the added benefits that would make the system stand out (wow factor).

In the third stage, the design and implementation teams collaborated with the product owner to generate the backlog of user stories to be implemented. Each story must result in an evaluable product. Then, a preliminary UX/UI design prototype was enacted as a visual aid to present features that directly correspond to the user stories. This UX/UI has been designed and presented using the platform Figma®, a collaborative design tool [22]. In addition, the design logic of the user's interactions with the system user case and package diagrams were created using the Unified Modelling Language (UML) so that the implementation team and product owner could document the process in a standard method. Finally, the prototype with the requirements and functional and technical feasibility was presented to the final user and product owner, who evaluated and provided feedback for final adjustments. Then, the implementation team analyzed the proposed solution, prioritized the stories, created group-related stories into epics as deemed necessary, and defined the minimum viable product (MVP) that provides the required usability, reliability, and functionality.

In the final stage, the structure, languages, tools, and resources are selected before the implementation sprint based on the functionality defined in the stories and the dynamic prototype. Then, written acceptance criteria associated with each user story were defined, and continuous testing was conducted throughout the implementation of the created stories into each sprint. For this purpose, Jira<sup>®</sup> was used as a collaborating Web project management tool, so the user stories with their acceptance criteria were placed in the backlog, the development road map in the Gantt plot, and the active assignments with incidences were monitored by sprints and development milestones of the collaborative work by using of its Kanban board [23]. The GitHub<sup>®</sup> platform was selected for version control and code integration [24]. In addition, Extreme Programming (XP) techniques were adopted to complement the Agile development methodology; specifically, the pair-programming technique was used because of the small size of the implementation team [25,26].

During each development sprint, an assignment is done in Jira<sup>®</sup>, and then specific stories are developed in parallel by the team employing different Git branches. Each programmer thoroughly tested each code in their current development branch by the acceptance criteria set by the product owner and the design team. When accomplished, it was reported in Jira<sup>®</sup>, where the progress tracking is available to the product owner. Epics and integrations are also tracked and tested before their inclusion in the primary production branch. Several development sprints were required to achieve the MVP.

### 3.2. Requirements and Techniques Definition

#### 3.2.1. Resource Management

As Web-based applications have been increasing in complexity and functionality, the paradigm of design and implementation has shifted from a monolithic approach, where a complete set of services must be developed, deployed, and scaled at the same time, using a shared codebase, which quickly becomes a great challenge as applications grow, into a design based on microservices.

A microservice strategy divides an application into a set of small services, each offering a subset of the functionality provided by the application. In this architectural design, every microservice can be developed and tested independently, utilizing different codebases, enabling each microservice's independent deployment, scaling, operating, and upgrading on a cloud computing solution [27].

Current software development practices have identified the benefits of a Microservices Architecture (MSA) in opposition to a monolithic design, as they allow for granular services that can be tightly integrated, providing benefits such as high availability, flexibility, scalability, and speed. Of particular importance, the flexibility of such a software architecture allows for the deployment on different computing environments, such as a public or private cloud, on-premises, or even a mixed approach: different components of the application can be installed on different computing environments without affecting the application functionality or usability. Cloud computing allows on-demand scalability of services, increasing or decreasing the number of computing resources based on the requirements of each software team, and depending on the specific design of the microservice, vertical or horizontal scaling can be achieved.

#### 3.2.2. Fuzzy Search Method

Information stored on the platform includes accompanying metadata, such as title, file description, and author. In addition, the registered users can visualize names, contact email addresses, and information on the module with which they are currently associated. The process of querying and retrieving relevant documents is known as Information Retrieval (IR) [28]. A UI text input element has been proposed to improve access to stored information that employs approximate string matching over the available metadata. This approximate string-matching for information retrieval functionality is provided by the software framework Fuse.js, which implements a variant of the bitmap algorithm,



also known as the Baeza-Yates-Gonnet algorithm [29]. This algorithm generates a list of matching items ranked by a relevance score. This score is determined by three factors: the Fuzziness score (FS), the Key weight (KW), and the Field-length norm (FLN). FS is based on inference rules based on the membership function associated with the Levenshtein distance described by Yujian et al. [30]. KW is a weight associated with the location of the found word in the user input, and FLN is associated with the inverse of the size of the field where the word was found. For example, the keywords field is smaller than the title field, but the summary is larger than the previous one [29].

### 3.2.3. Multidimensional Data Visualization and Statistics Characterization

Data visualizations play a relevant role by allowing analysts to uncover hidden patterns and correlations within datasets, fostering new perspectives and deep comprehension. Furthermore, these graphics effectively convey complex concepts to varied audiences, from experts to novices, enhancing accessibility and clarity [31,32]. As mentioned by Schmidt et al., the essential charts for visualizing data encompass scatter plots, line plots, area plots, bubble charts, bar charts, pie charts, and doughnut charts, so all these plots are implemented in Python [33]. Once equipped with these suitable tools, the subsequent focus lies in including multivariable visualization techniques.

The interpretation and understanding of data are reliant on the selection of appropriate visualization techniques. With many available methods, determining the most suitable methodology for a specific dataset or decision-making process can be challenging. The correct choice directly impacts how the data is interpreted and comprehended by those analyzing it. In increasingly large and complex datasets, the need for scalable visualization approaches capable of handling the quantity and intricacy of information becomes even more pertinent. Methods such as PCA and t-SNE emerge as essential solutions to address the visualization of extensive data dimensions.

One option is the PCA, which identifies the most significant basis for representing a dataset [34]. This transformation to a new basis set unveils the dataset's underlying structure and filters out noise. So, it is used to reduce dimensionality, data compression, feature extraction, and data visualization.

Another option is t-SNE, a dimensionality reduction technique widely employed in ML and data visualization. Its primary objective entails projecting high-dimensional data into a lower-dimensional space while preserving inherent relationships and similarities between data points. Even though t-SNE is not a bona fide classifier, it finds extensive application in complex data analysis and visualization tasks [35].

### 3.2.4. Knowledge-Sharing Implementation Tools

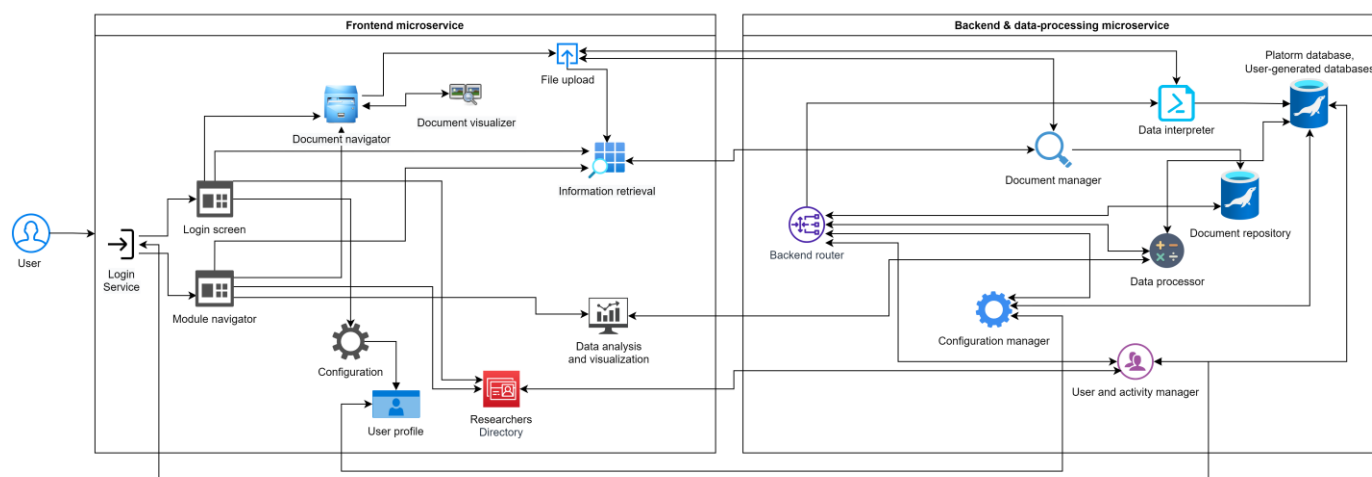
The knowledge-sharing promotion and the need to create a specific solution for the researcher's community that generally uses several commercial software or platforms for collaboration were considered goals to achieve. It was essential to involve the final user in defining the requirements during the design and implementation. Constant interaction in Design thinking (DT) in the creation process helps minimize their learning curve and increase product acceptance. This methodology was proposed by the International Business Machines Corp. (IBM) [36], and it is considered a framework for software design that covers a set of good practices of software platform development when it is combined with Agile development techniques such as Scrum and Kanban for the system implementation [37]. The SCRUM methodology promotes continuous collaborative work of the development team (product owner, designers, and programmers) with frequent final user feedback based on iterative cycles called sprints. Agile implementation is carried out through planning meetings, assignment of activities, monitoring, delivery, and reviews of the developed artifacts. At the end of each sprint, executable deliveries are presented to the product owner and final user [25].

### 3.3. System Implementation

Implementing the proposed methodology to create a platform that enables researchers in a multidisciplinary project to share information and research artifacts concisely calls for developing a purpose-specific data management tool. Access to the system is provided through a Web application because up-to-date Web browsers supporting current technologies have become ubiquitous across all principal desktop and mobile operating systems. In addition, a Microservices Architecture (MSA) was selected because of its granular services that can be tightly integrated, providing benefits such as high availability, flexibility, scalability, and speed.

The proposed platform implements role-based access control (RBAC) employing discretionary access control, where a role hierarchy determines the amount of control a user has over the data they produce or manage. This type of system enables the privacy and confidentiality of user information at each step of the research process [38].

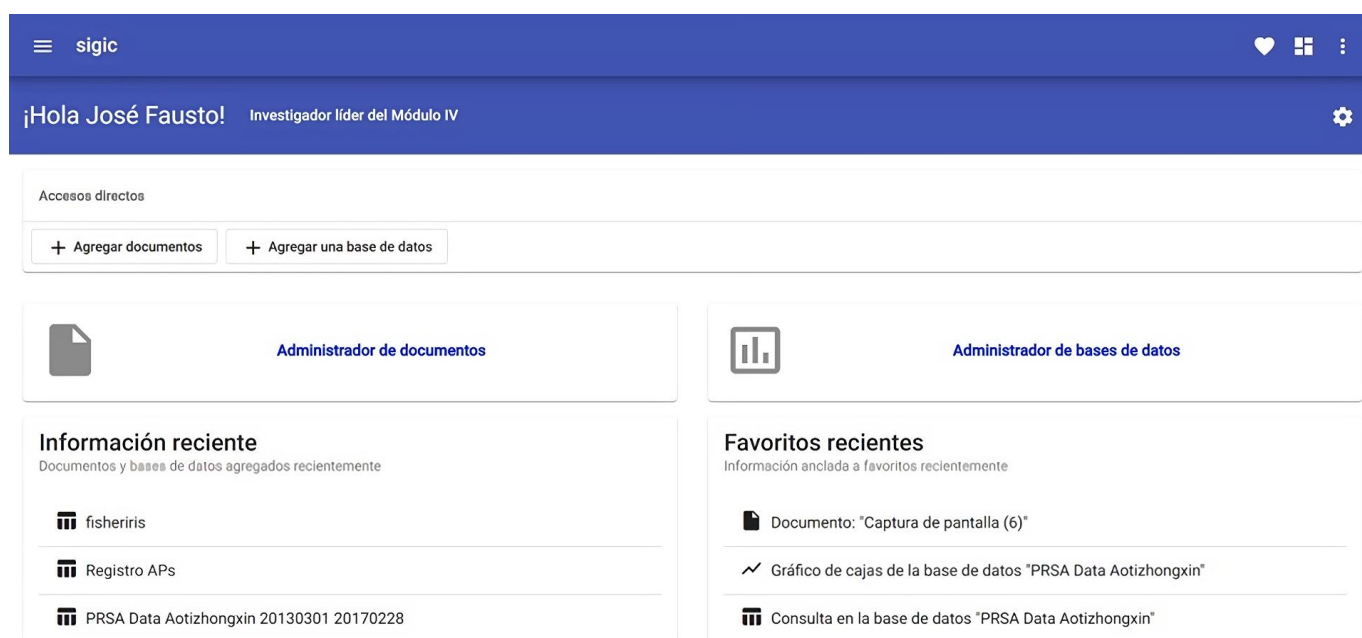
The proposed platform consists of two interacting software components, whose interactions can be visualized in Figure 1.



**Figure 1.** Frontend and Data-processing backend components and interactions. The arrows represent directed information flux.

- **Frontend.** This component is a microservice that generates and serves the Web application to every connected client. This microservice has been designed to support horizontal scaling, as it only serves as a stateless proxy between the user interface and the data-processing backend.
- **Data-processing backend.** This component is a microservice that attends user requests through a Representational State Transfer (REST) Application Programming Interface (API). It handles information, uploads, data storage, analysis, and processing. Communication with the Relational Database Management System (RDBMS) is provided using two distinct Object Relational Mapping (ORM) interfaces. The first ORM handles the platform's data and state, while the second handles creating and manipulating user-generated databases. This microservice has not been designed with thread-safety methods. Since information processing is handled stateful, it is currently only possible to support the vertical scaling of this microservice, which means increasing resources for processing the information.
- **Relational Database Management System.** This component is external to the application and can be implemented as either a vertically scaled physical or virtual server or a cluster of vertically scaled load-balanced servers to enable horizontal scaling. The platform employs a MySQL-compatible database engine with two information schemas: the first one stores the platform's data, which considers the users and document metadata and pointers, and the second one stores raw user databases.

- File storage backend. This component is external to the application and has been implemented as a simple storage service (S3) that does not require any external cloud-dependent API; instead, the file storage backend stores flat (that is, non-hierarchical) data identified by their UUID, as described in [39,40], whose metadata is stored in the platform database.
- User authentication and login process. Access to the platform is provided by a user-password authentication mechanism, where users and passwords are stored in the application database using the PKCS2 cryptography algorithm, as detailed in Moriarty et al. [41]. Every time a user is successfully authenticated into the platform, the current date and time in UTC format are stored in the database in a dedicated “last login” field. This field is used to determine if a user has not logged into the application since the creation of the account, in which case the entry for the user is deleted after one day using the periodic execution of a background housekeeping task. Upon successful authentication to the platform, the user dashboard is displayed. The user dashboard contains quick links to recently uploaded and anchored information through a pointer to a document or database, as shown in Figure 2.

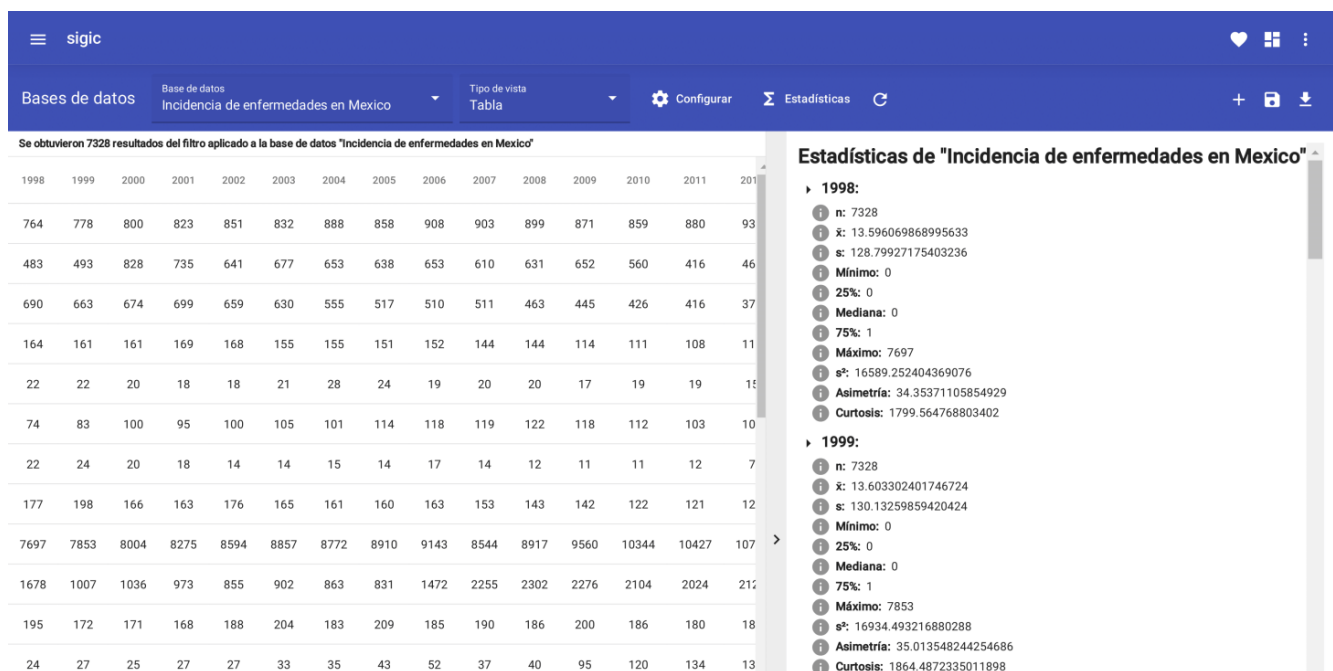


**Figure 2.** User dashboard in the collaborative platform.

- Data analysis and visualization component. This component allows users to upload tabular data in several standard formats, which are then interpreted by the data-processing backend and stored as intermediate tables in the database backend. From the user interface into the databases section, users can select different datasets from the dropdown menu. Three data display methods exist: previsualization, tabular display, and graphical format.

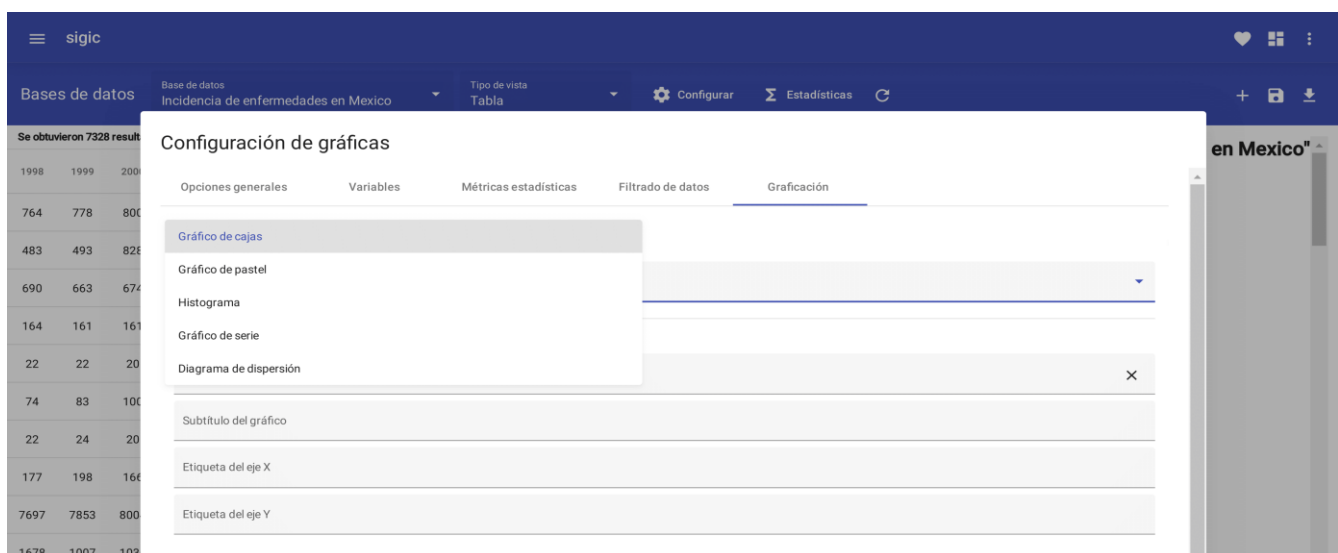
Previsualization shows the first twenty elements of the dataset and is the defined option every time a new dataset is selected. The table form enables the use of each data analysis toolkit option built on the platform. This option lets the user exclude columns from the datasets, filter information based on class or numeric value according to a set of rules that can be configured and choose from a set of basic statistical descriptions. The statistical descriptions included a configurable set of  $n$  quantiles provided by default quartiles, deciles, and centiles. It also shows the minimum, median, maximum, and sample statistics, such as average, variance, standard deviation, skewness, and kurtosis, as shown in Figure 3.





**Figure 3.** Previsualization in the database management dashboard with descriptive statistics panel.

In the graphical format option, the tabular data are presented for visual analysis through various data visualization and processing functionalities identified as valuable to researchers in the DT process. These include the creation of box plots, pie charts, histograms, multi-series line charts, and dispersion charts, the last one utilizing either t-SNE or PCA. The user is assisted with a brief explanation of the method and possible results to select the data visualization that best fits the context of their data utilizing the graph configuration component, as seen in Figure 4.



**Figure 4.** Graph configuration component.

- **Document manager.** This component allows users to upload files of arbitrary format. However, some Web-supported multimedia formats (images and PDF documents) can be previewed directly on the platform, either in the right-side panel UI element or in a new browser tab or window. These previews are not downloaded onto the user's computer but rather cached in memory. As previously described, a text box UI

element is provided at the top of the screen to provide search capabilities using fuzzy matching to assist the system use.

- **Researcher directory.** This component allows the user to search for the contact information of a specific researcher, either by their name or current association with a module. In the same fashion as the document manager, a text box element is provided at the top of the screen to provide search capabilities employing fuzzy matching, as previously described.

### 3.4. System Testing and Feedback

As the methodology proposed, the testing and feedback process was divided into three sequential levels:

1. **Test by Programmers in Production:** The programmers evaluated the expected operation in the production platform, verifying the proper integration of the system.
2. **Testing by Project Members:** The product owner and two members not involved in the design and implementation of the system reviewed compliance with requirements and usability in the final Web version.
3. **Testing by Module Researchers:** Experts in specific modules validated functionality and suitability for their specialized needs and feedback in the Web version.

#### 3.4.1. Project Members' Test

This test evaluated the performance and operability of the system as well as the accomplishment of the acceptance criteria. Testers had no prior knowledge of architectural design and had not been involved in developing the underlying code. Each test stage was carefully documented in a traceability matrix in a test log. It registers in detail each component or segment of the system evaluated and its specific location. The actions and events carried out during the tests were recorded, as well as the observations and conclusions drawn by the evaluators. In case of failures or malfunctions, details of the nature of the problem and related circumstances were documented. These records also provided essential information to identify areas with potential for improvement and optimization, so the programmers' team discussed these logs. A critical aspect of this evaluation was the assignment of scores to measure the impact of the specific actions carried out in the system. These scores were based on a five-level Likert scale, ranging from 1 (indicating difficulty or low ease) to 5 (indicating high ease). This rating made it possible to quantify the user experience and provided valuable data to understand the level of ease and effectiveness of the actions within the system.

Finally, the product owner tested the system's usability and performance using the Fisher Iris dataset [40] and three multivariable datasets with 14 variables and instances quantity classified as small, medium, and large data.

#### 3.4.2. Testing by Researchers of a Multidisciplinary Project

The system was evaluated monthly, showing it to the researchers considered our final users. They were given a series of tasks to have homogeneous control of their challenges and were asked to answer a questionnaire. They cover the actions associated with the added functionality in the final system. After answering the questionnaire, they were given free access and freedom to use the platform. The programmers' team evaluated the feedback concordance with the predefined hills, and then priority and improvement feasibility were defined. The exercises requested from the researchers were: 1. Enter the platform using the link; 2. Create a new account; 3. Log in with the username and password; 4. Upload three files in different formats; 5. View details by clicking on a file; 6. Mark 3 files as favorites; 7. Access the favorites section in the document browser; 8. Change the name and description of a file; 9. Download a file from the favorites section; 10. Delete a file from "My Documents"; 11. Delete all files; 12. Return to the main board; 13. Add a database; 14. Select the added database; 15. Activate the statistics panel; 16. Change the general settings in "Options"; 17. Change the variable configuration in the configuration;

18. Save a query as a favorite; 19. Change statistical metrics settings; 20. Add two filters in the data filtering; 21. Save another query as a favorite; 22. Download a table (optional format); 23. Download statistics (optional format); 24. Request and customize a boxplot; 25. Save the chart as a favorite; 26. Customize and request a pie chart; 27. Customize and request a histogram; 28. Save another chart as a favorite; 29. Customize and request a serial chart; 30. Customize and request a trending series chart; 31. Save the chart as a favorite; 32. Download the graph as an image; 33. Add another database; 34. Go to “Favorites”; 35. Identify details of a favorite; 36. See a favorite in menu options; 37. Edit details of a file in favorites; 38. Remove a file from favorites; 39. Go back to the main board; 40. Go to the user’s profile from the navigation bar; 41. Make changes to the general information; 42. Request password change; and 43. Sign out.

This meticulous and structured approach to testing ensured a comprehensive, objective, and detailed evaluation of the system from various perspectives. The logbook became a resource for monitoring, analysis, and continuous system improvement as it progressed through development and neared final implementation. In addition, and more importantly, considering the objective of the methodology, the researcher, as a final user, was introduced gently to the system’s functions and became used to browsing the system. So, a questionnaire was applied after carrying out the 43 exercises in the SIGIC platform.

#### 4. Results

The result of the implementation of the methodology is a Web platform that provides multidisciplinary researchers with an essential data analysis toolkit built-in in the platform Scientific Information Management System (SIGIC-Sistema de Gestión de Información Científica) hosted in “<https://sigic.labips.com.mx/> (accessed on 20 August 2023)” as a product of the interaction of multidisciplinary teams in the project named: multidimensional models of temporal series associated to the anthropic contamination in marine organisms consumed by humans and its effect on their overall health, funded by IPN.

##### 4.1. Researcher’s Test

In this section, the questionnaire results are summarized. Creating a new account was reported as very easy (66.7%) and easy (33.3%). Entering the created account, uploading files, saving a document in favorites, and identifying the type of files in the Documents section were evaluated as very easy (88.3%) and easy (16.7%).

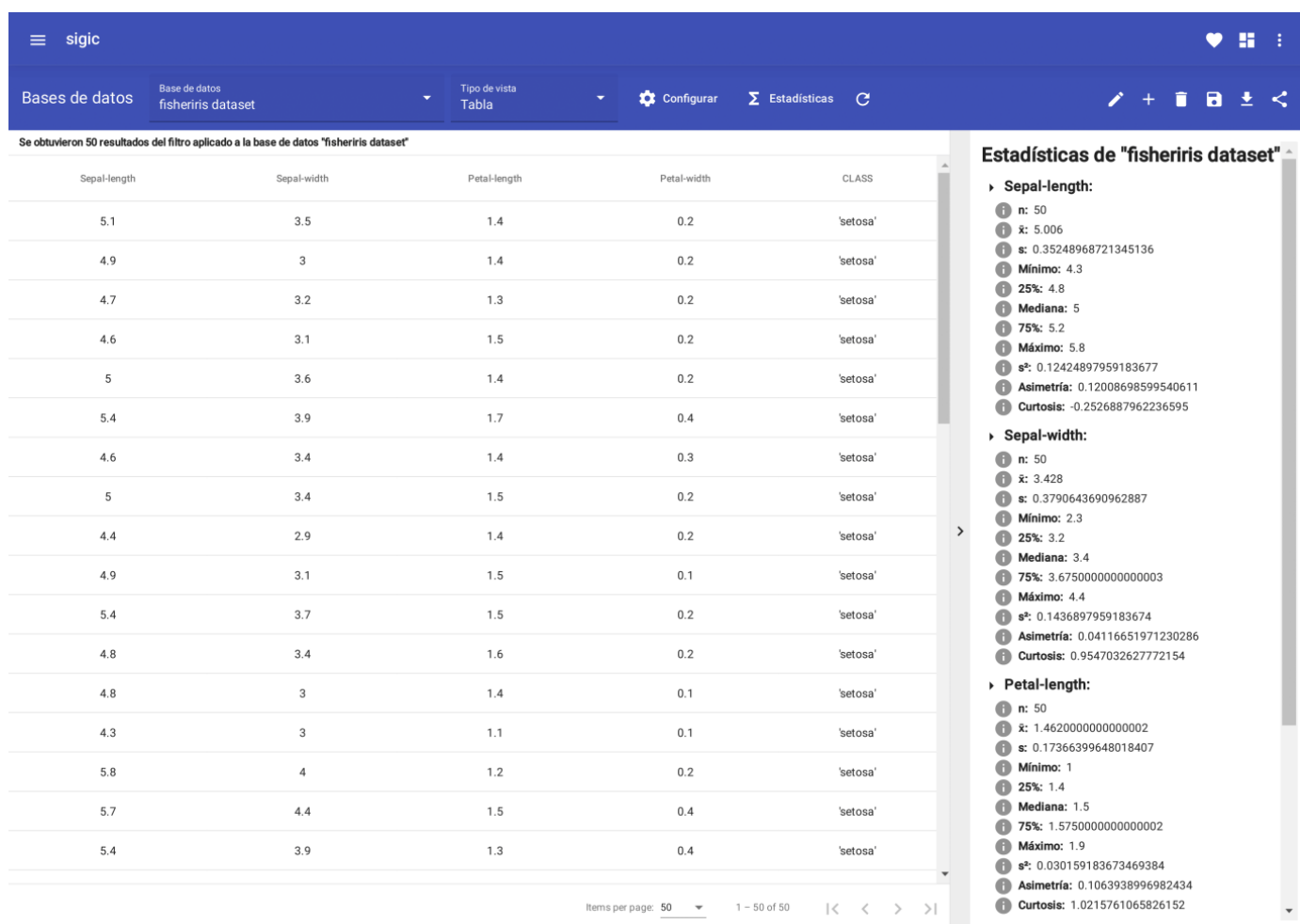
User feedback about the system was mainly positive, although it is considered that practice and knowledge of the multivariable visualization methods are required to take full advantage of it. It is noted that the account verification email is sometimes sent to the spam folder in the activation process. A key observation is the difficulty encountered by some researchers when starting a user account with a lowercase letter, unaware of this specification.

In general, they report that it was intuitive and easy to use. However, an aspect for improvement was pointed out: the mismatching password input with the previously registered lack of a warning message. Some users faced issues while accessing the system, but they could manage the task after receiving guidance. They highlight positive aspects such as helpful descriptions and the ability to drag and select files efficiently. In addition, the tool allows users to download files and statistics in different formats, although some have reported difficulties accomplishing this task. As for the graph options, users found them easy to use, although in some cases, they were unable to create graphs or identify how to apply specific settings.

In summary, most people found the tool intuitive and valuable, though some others found areas for improvement in clarifying error messages and identifying specific features. In the end, the critical question is, “Would you consider using SIGIC in a project?” 83.3% answered Yes, while 16.7% said No.

#### 4.2. Product Owner Test

The product owner used the Fisher Iris dataset for testing. A data filter was applied to select the Seratosa class, and the results in SIGIC are shown in Figure 5. Then, a box plot of this class was generated and displayed in Figure 6. The sample statistics of the Seratosa class were obtained and presented as a PDF file report and shown in Figure 7; the results are according to the reported mean values from Fisher's work [42]. The boxplot PDF file of the data was created and downloaded, as shown in Figure 8.



**Figure 5.** Previsualization data and Statistical results obtained from 50 Setosa class samples generated by SIGIC in the dashboard.

These results show the platform's accuracy and export capabilities, ensuring the fidelity of the generated outcomes and the seamless conversion to PDF format while retaining the associated dataset metadata. The box plot was selected as a visual representation to provide a comprehensive overview of the distribution and variability inherent in each feature associated with the Seratosa class. The box plot offers insights into central tendencies, spread, and potential outliers within the data.

Finally, due to the four features of each class of the Iris dataset, transformed representations were constructed for the four variables (sepal length, sepal width, petal length, and petal width) in a two-dimensional space using PCA and t-SNE techniques, as depicted in Figures 9 and 10.

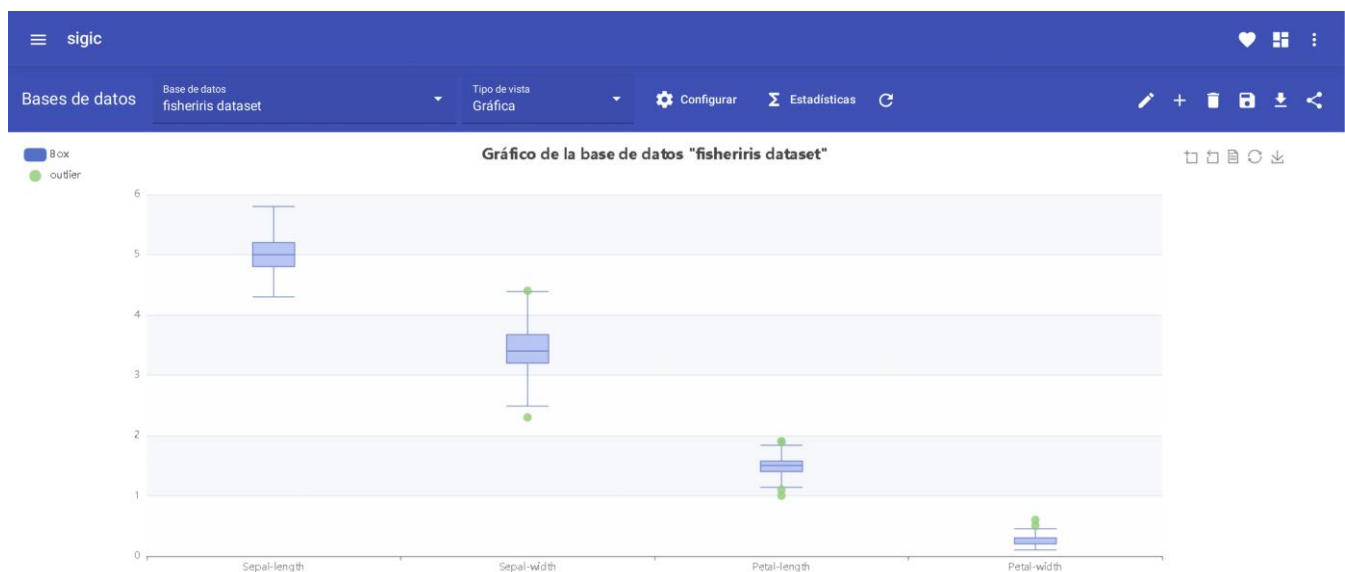


Figure 6. Box plot generated in the SIGIC platform of the data for the Seratosa class.

## fisheriris dataset

The dataset contains information about three different species of iris: Iris setosa, Iris versicolor, and Iris virginica. The dataset consists of 150 samples, 50 from each species of iris.

### Query parameters

#### Selected variables:

- Sepal-length
- Sepal-width
- Petal-length
- Petal-width
- CLASS

#### Filters applied:

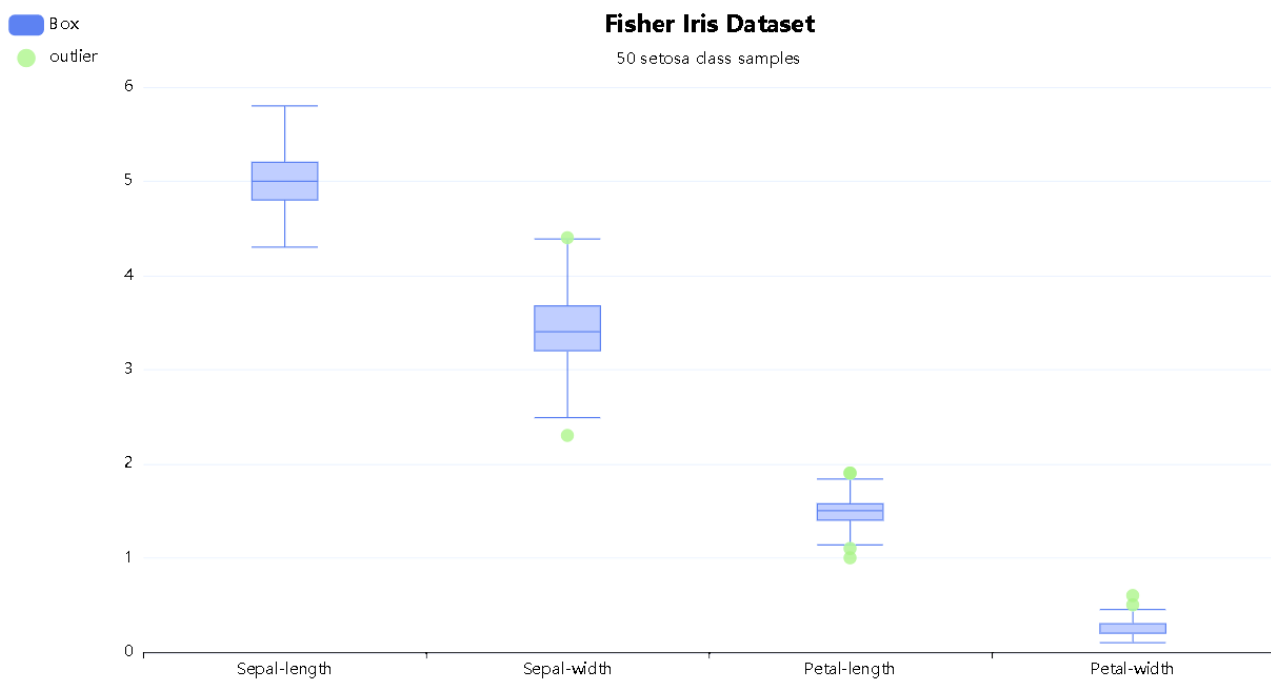
1. Variable name: CLASS, Operation: eq('setosa')

	Sepal-length	Sepal-width	Petal-length	Petal-width
count	50.000000	50.000000	50.000000	50.000000
mean	5.006000	3.428000	1.462000	0.246000
std	0.352490	0.379064	0.173664	0.105386
min	4.300000	2.300000	1.000000	0.100000
25%	4.800000	3.200000	1.400000	0.200000
50%	5.000000	3.400000	1.500000	0.200000
75%	5.200000	3.675000	1.575000	0.300000
max	5.800000	4.400000	1.900000	0.600000
var	0.124249	0.143690	0.030159	0.011106
skew	0.120087	0.041167	0.106394	1.253861
kurt	-0.252689	0.954703	1.021576	1.719130

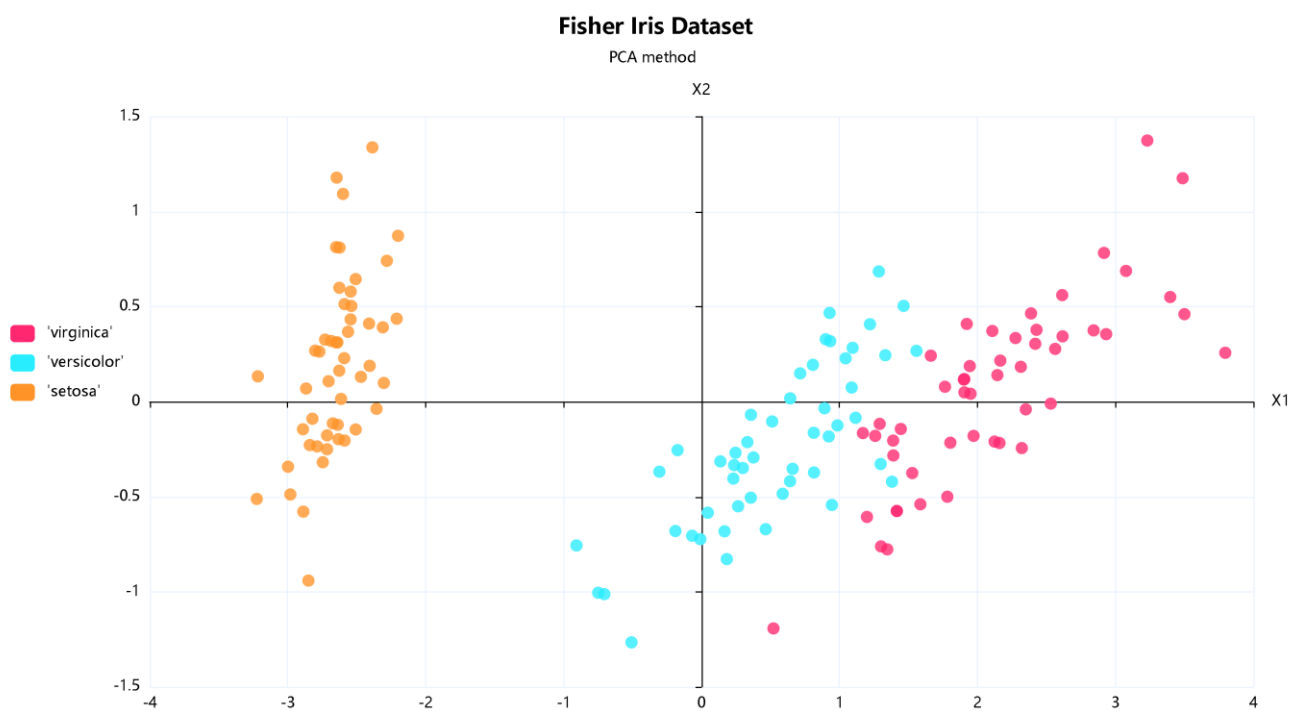
Data obtained from SIGIC, the scientific management system designed by the National Polytechnic Institute of Mexico at the Laboratory of Instrumentation and Signals Processing (LabIPS UPIITA-IPN).

Figure 7. Statistical results in PDF format obtained from 50 Seratosa class generated by SIGIC.

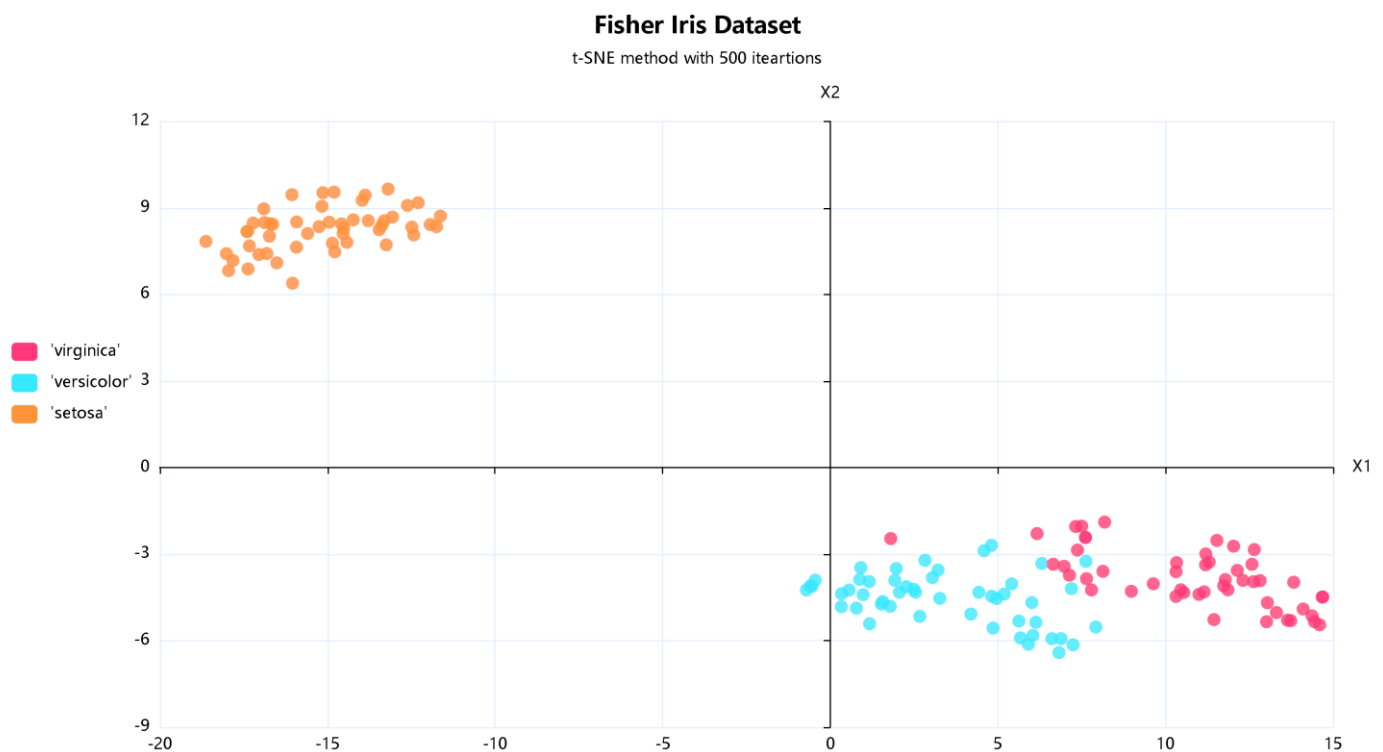




**Figure 8.** The downloaded box plot image generated in the SIGIC platform for the Seratosa class.



**Figure 9.** The scatter plot of the Fisher Iris Dataset reconstructed with the first two principal components of the PCA transform.

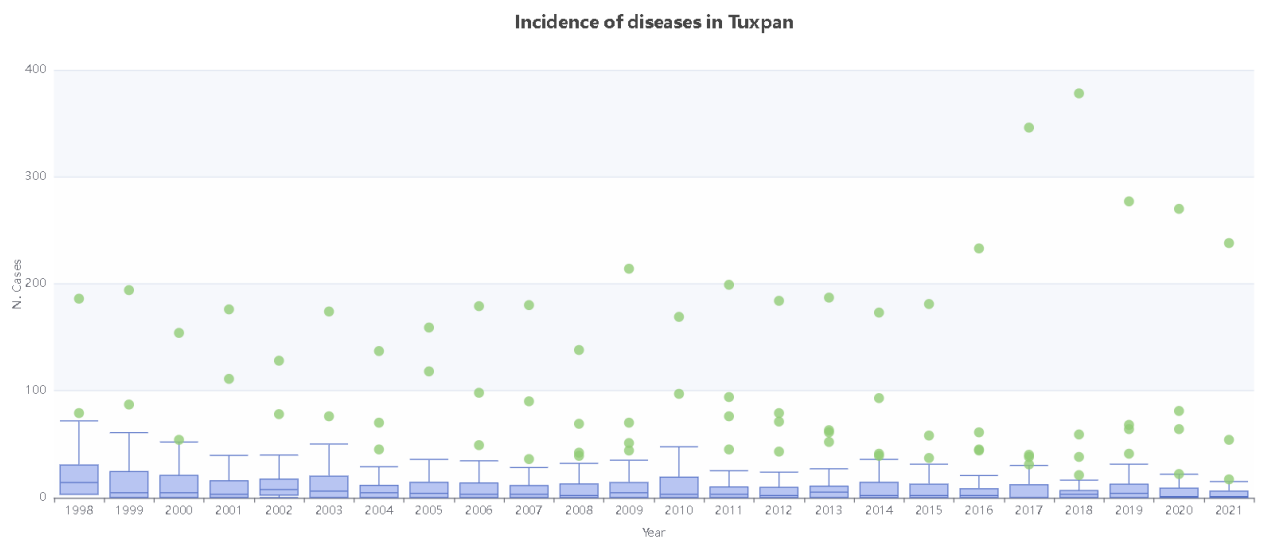


**Figure 10.** The scatter plot with the values resulting from the t-SNE method is based on the features associated with four classes defined in the Fisher Iris database.

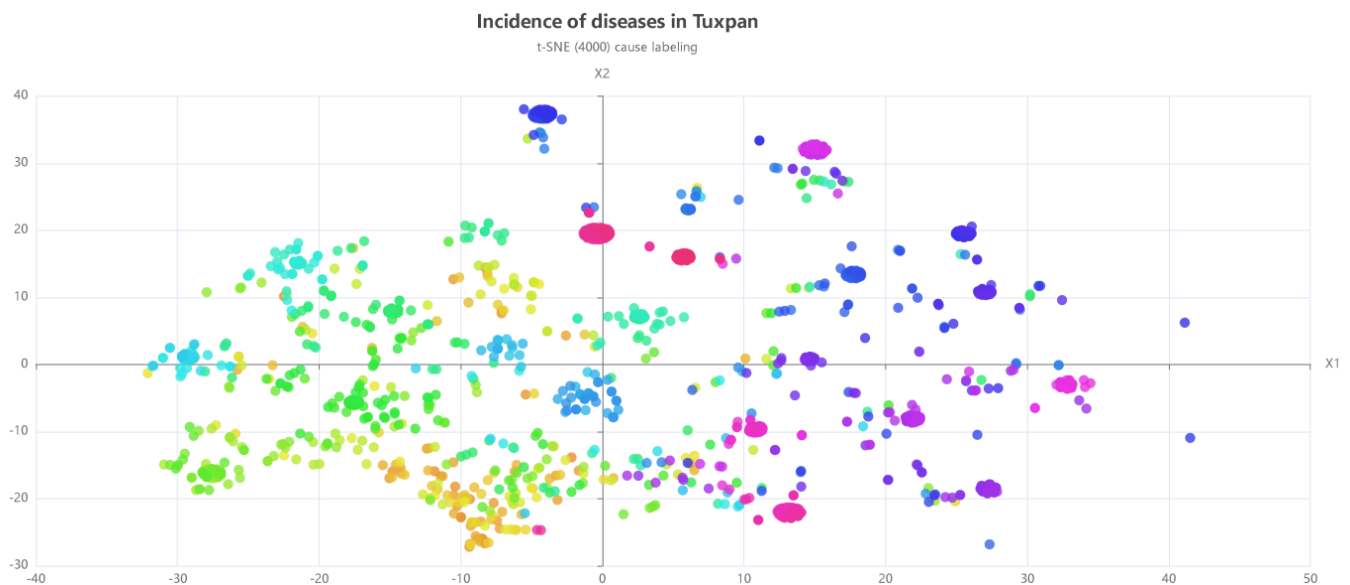
#### 4.3. Case of Study

Within the framework of a multidisciplinary approach, disease records were examined both at the local level in Tampamachoco (instances), Veracruz state (instances), and the national level (instances). This analysis was conducted to test the scalability performance. These data were provided as partial results of the multidisciplinary studies in environmental health supported by the SIGIC.

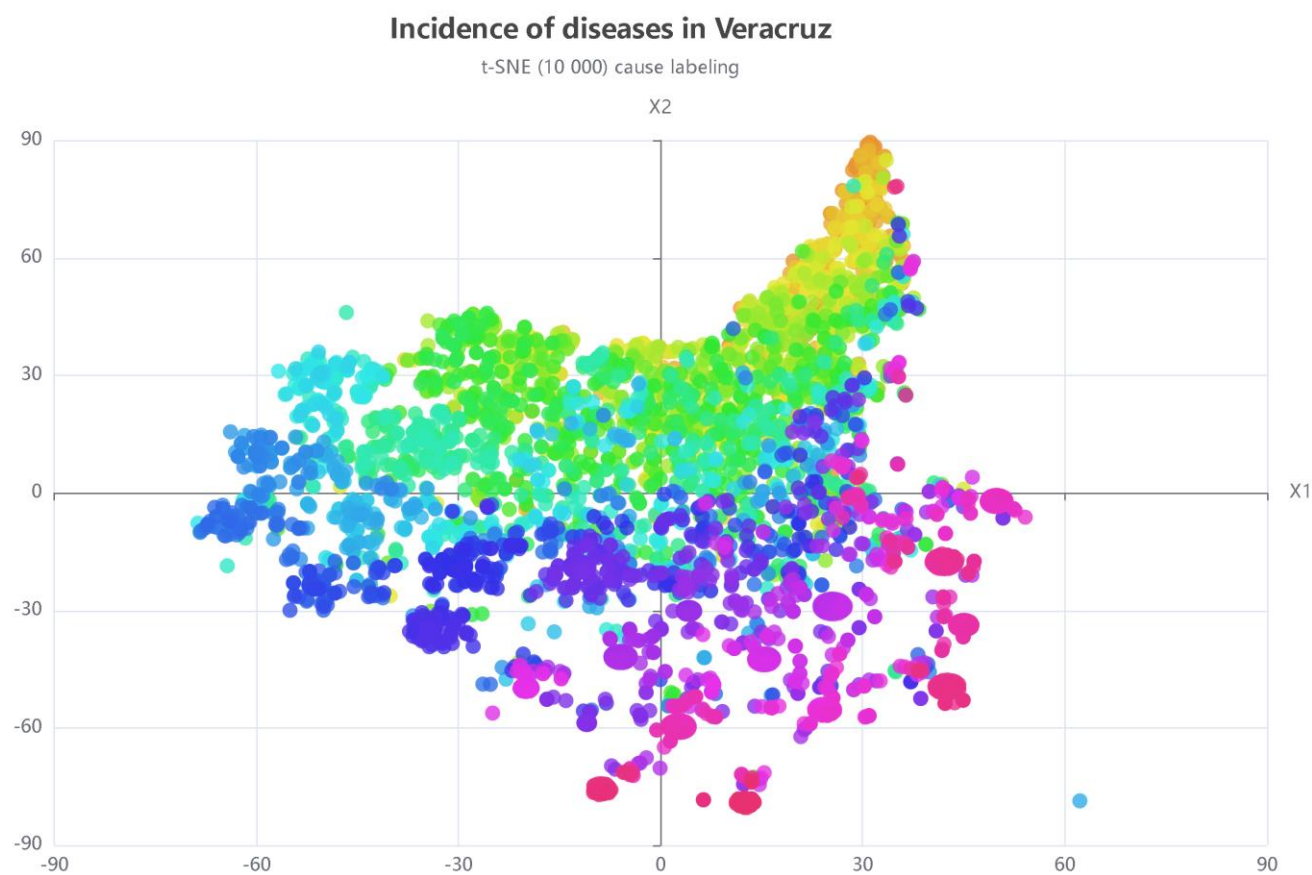
The data utilized in this study were obtained from the website of the Dirección General de Información en Salud (DGIS, General Direction of Health Information) [43]. These data are from 1998 to 2021 and correspond to the incidence value of 1246 diseases in the mortality reports. Figure 11 depicts a box plot illustrating the yearly median cases for various categorized causes, including upper quartiles, maximum values, and outliers for the Tuxpan report. Figure 12 displays a scatter plot generated through t-SNE, highlighting how diseases are distributed over time. Values closer to the origin indicate a lower frequency over time, whereas grouped diseases reveal a more intense temporal relationship. Notably, due to the volume of categorized cases, the color palette exhibits subtle variations. The color assignment is made by dividing the six-digit hexadecimal system by the number of diseases, excluding black and white. The results extend to the data from Veracruz, and at the national level, they are presented in Figures 13 and 14, respectively. These three plots required different resource availability; even ten thousand iterations were computed in each case. The number of elements for each class was 1246, 4461, and 7328, so the instances quantity increased from the initial number to 358% and 588%, respectively.



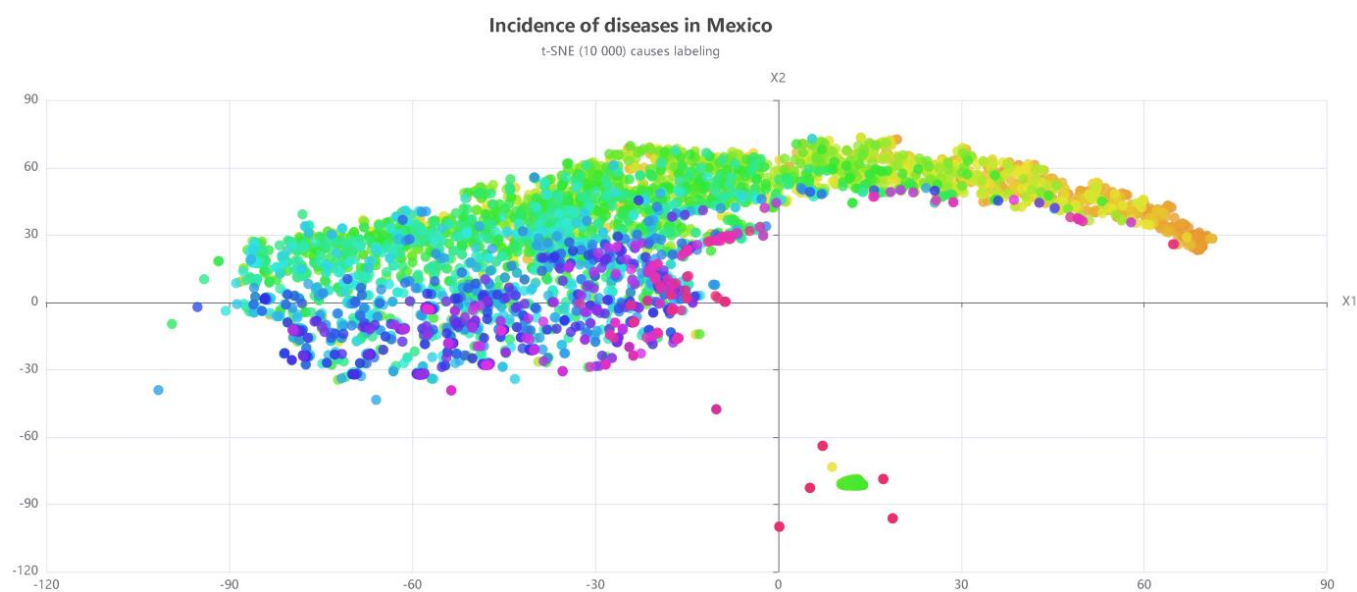
**Figure 11.** Box plot of disease incidences in Tuxpan. Number of cases versus year for 1246 different causes.



**Figure 12.** Scatter plot of disease incidences in Tuxpan for 1246 different causes from 1998 to 2021 using t-SNE at 10,000 iterations. This image presents clusters generated by the correlated temporal bivariable structure of the dataset. Each dot is a disease, and the color assigned is according to the closeness to the group's centroid.



**Figure 13.** Scatter plot of disease incidences in Veracruz for 4461 different causes from 1998 to 2021 using t-SNE at 10,000 iterations. This image presents clusters generated by the correlated temporal bivariable structure of the dataset. Each dot is a disease, and the color assigned is according to the closeness to the group's centroid.



**Figure 14.** Scatter plot of disease incidences in Mexico for 7328 different causes from 1998 to 2021 using t-SNE at 10,000 iterations. This image presents clusters generated by the correlated temporal bivariable structure of the dataset. Each dot is a disease, and the color assigned is according to the closeness to the group's centroid.

## 5. Discussion

This paper presents a combination of Design Thinking and Agile problem-solving techniques that allows a design team to study the workflow of its intended audience, identify flaws, roadblocks, or means of improvement, and come up with brainstorming techniques that aim to reduce the shortcomings and improve the overall experience of the work process. Considering the specific needs of the researcher communities in knowledge-sharing, this methodology based on design thinking has considered the intrinsic factors involved in the intention to share knowledge. In addition, the SCRUM approach using digital project management promotes continuous collaborative work of the development team (product owner, designers, and programmers) with continuous final user feedback based on iterative cycles called sprints. Agile implementation through planning meetings, assignment of activities, monitoring, delivery, and reviews of the developed artifacts is currently a well-supported method [44].

In contrast to a monolithic version, this microservices-based platform enables both horizontal and vertical scaling of services, which provides the capabilities for growing or shrinking computing resources based on the actual requirements of each specific multidisciplinary team, as was shown in the last test. Furthermore, as the system has been designed with a cloud-agnostic architecture, deployment is not constrained to any specific cloud computing service provider. It can thus be deployed in public or private clouds, hybrid environments, or utterly standalone on dedicated hardware, as the research project leaders deem appropriate.

Medium to large-scale research in environmental health studies calls for a multidisciplinary approach, where several highly specialized research teams should collaborate in each step of the data analysis process to promote a holistic analysis. This methodology helps to manage the data lifecycle and research artifacts, considering that different research teams might use techniques and tools unique or specifically tailored to their fields of study, so developing this data management and analysis system for the intermediate sharing of reports and datasets in a platform available across all teams has the potential of reducing the amount of information reinterpretation and reformatting, thereby streamlining the workflow of the overall research process. Our study shows that knowledge-sharing should be focused on the intermediate process associated with the dataset and its expert interpretation instead of trying to have all the highly specialized tools available for everyone. Moreover, even previous works [17,20,26,27] reported challenges in disseminating diverse visualization techniques, limited awareness of graphic utilization, and the absence of tools with advanced methods. Some essential tasks could be quickly performed with high-level software such as MATLAB®, Mathematics®, Prism®, and Stata®, but all of them require a commercial or academic license. Some powerful open-source software, such as Phyton, has a long learning curve, and researchers do not always want to invest time or money in learning new software because they have specialized software.

This study aligns with the evidence in the literature [19–21], pointing out the potential of t-SNE as a 2D graphical representation of a multivariable dataset despite its insufficient dissemination. Similarly, various systems provide advanced tools, but their low availability at academic levels diminishes their promotion. According to the method used, the axes represent the temporal relationship between all diseases in two variables and the given interpretation is that if the instance is closer to zero, it implies that it has little occurrence, and the farther from the origin, the greater the occurrence. In addition, diseases that are closer to each other imply that they had a higher correlation over the recorded time.

Furthermore, in the context of SIGIC, methods for histogram generation remain widely unknown, even among researchers, so the included help tips and available references are reported as useful. The study case showed one of the topics of interest for this multidisciplinary project directed to analyze environmental effects on human health, encompassing a set of statistical calculations, box plots, and scatter diagrams that provide an overarching view of the incidence and propagation of diseases over time in the investigated regions. Interpreting the result is not the primary goal of this study, but this approach highlights



significant patterns and relationships within the dynamics of the identified causes of mortality. This platform is intended to allow collaboration and promotion of sharing in the multidisciplinary researchers' teams.

## 6. Conclusions

The SIGIC Web platform was created using the proposed hybrid methodology that provides an understanding of experts' workflow and techniques.

The hybrid methodology found an effective way to overcome the diversity of sources and tools required in each module and propose to share, after preprocessing, the information, creating a context of interaction and visualization where the expert could share their findings and obtain feedback from the other interdisciplinary team members after they visualize the datasets or reports in the platform. Researchers reported that SIGIC broadens their perspectives. They outline that it promotes knowledge sharing, covering basic everyday activities in a researcher's workflow. Even though the researchers' feedback in Section 4.1. showed a general acceptance, and further studies are required on the impact of age, gender, culture, organizational structure, and use of technology on effective knowledge-sharing.

Therefore, in future work, on the one hand, additional functionalities could be included to reinforce the researchers' acceptance. On the other hand, a longitudinal study of the multidisciplinary group activities in the platform could help gain a comprehensive understanding of the user motivations and propose further tools to make this more attractive or helpful in knowledge-sharing.

## 7. Patents

Sistema de Gestión de información científica (SIGIC) is a copyright version registered in National Institute of Copyright (INDAUTOR) with number, 03-2023-030112290400-01 by Instituto Politécnico Nacional on 3 March 2023.

**Author Contributions:** All authors have equally contributed to this work. L.I.G.-J.; conceptualization, project administration, resources, supervision, original draft, revision, validation, J.F.R.-L.; data curation, software, revision, validation, writing, A.S.-H.; software, writing and editing, B.T.-C.; validation, review and editing, P.G.-M.; conceptualization, methodology supervision, review, M.F.M.-R.; formal analysis, review and editing. All authors have read and agreed to the published version of the manuscript.

**Funding:** Instituto Politécnico Nacional funded this work through the project "Multidimensional models of temporal series associated to the anthropic contamination in marine organisms consumed by humans and its effect on their overall health" with annual grant numbers SIP 20211164, SIP 20220701 and SIP 20230872 within 2021–2023.

**Data Availability Statement:** The Iris data presented in this study are openly available in the Iris—UCI Machine Learning Repository at <https://archive.ics.uci.edu/dataset/53/iris> (accessed on 20 August 2023). The preprocessed Mexican mortality data presented in this study are available on request from the corresponding author. The data are not publicly available due to the current use by the multidisciplinary teams.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Shaoshao, X.Y.; Wu, J.; Deng, C.; Li, P.G.; Feng, C.X.J. A web-enabled collaborative quality management system. *J. Manuf. Syst.* **2006**, *25*, 95–107. [\[CrossRef\]](#)
2. Yoon, S.W.; Matsui, M.; Yamada, T.; Nof, S.Y. Analysis of effectiveness and benefits of collaboration modes with information and knowledge-sharing. *J. Intell. Manuf.* **2011**, *22*, 101–112. [\[CrossRef\]](#)
3. Akhavan, P.; Rahimi, A.; Mehralian, G. Developing a model for knowledge sharing in research centres. *Vine* **2013**, *43*, 357–393. [\[CrossRef\]](#)
4. Lubell, M. Collaborative partnerships in complex institutional systems. *Curr. Opin. Environ. Sustain.* **2015**, *12*, 41–47. [\[CrossRef\]](#)
5. Franz, M.; Lopes, C.T.; Huck, G.; Dong, Y.; Sumer, O.; Bader, G.D. Cytoscape.js: A graph theory library for visualisation and analysis. *Bioinformatics* **2016**, *32*, 309–311. [\[CrossRef\]](#) [\[PubMed\]](#)

6. Van Rossum, G.; Drake, F.L., Jr. *The Python Language Reference*; Python Software Foundation: Wilmington, DE, USA, 2014.
7. Singhal, S.; Jena, M. A study on Weka tool for data preprocessing, classification, and clustering. *IJITEE* **2013**, *2*, 250–253.
8. Kulkarni, E.G.; Kulkarni, R.B. Weka, powerful tool in data mining. *IJCA* **2016**, *975*, 8887.
9. Attwal, K.P.S.; Dhiman, A.S. Exploring data mining tool Weka and using Weka to build and evaluate predictive models. *Adv. Appl. Math. Sci.* **2020**, *19*, 451–469.
10. Mullon, P.A.; Ngoepe, M. An integrated framework to elevate information governance to a national level in South Africa. *Rec. Manag. J.* **2019**, *29*, 103–116. [[CrossRef](#)]
11. Janssen, M.; Brous, P.; Estevez, E.; Barbosa, L.S.; Janowski, T. Data governance: Organising data for trustworthy Artificial Intelligence. *Gov. Inf. Q.* **2020**, *37*, 101493. [[CrossRef](#)]
12. Taivalsaari, A.; Mikkonen, T.; Ingalls, D.; Palacz, K. Web Browser as an Application Platform. In Proceedings of the 2008 34th Euromicro Conference Software Engineering and Advanced Applications, Parma, Italy, 3–5 September 2008; pp. 293–302.
13. Taivalsaari, A.; Mikkonen, T.; Pautasso, C.; Systä, K. Comparing the Built-In Application Architecture Models in the Web Browser. In Proceedings of the IEEE International Conference on Software Architecture (ICSA), Gothenburg, Sweden, 3–7 April 2017; pp. 51–54.
14. Waseem, M.; Liang, P.; Shahin, M. A Systematic Mapping Study on Microservices Architecture in DevOps. *J. Syst. Soft.* **2020**, *170*, 110798. [[CrossRef](#)]
15. Qin, X.; Luo, Y.; Tang, N.; Li, G. Making data visualisation more efficient and effective: A survey. *VLDB J.* **2020**, *29*, 93–117. [[CrossRef](#)]
16. Ou, J.; Zhu, L.J. trackViewer: A bioconductor package for interactive and integrative visualisation of multi-omics data. *Nat. Method* **2019**, *16*, 453–454. [[CrossRef](#)] [[PubMed](#)]
17. Morales, J.; Echavarría, F. Methodology to explore open data of road crashes using Data Science: Case Medellín. *Ingeniare* **2019**, *27*, 495–509. [[CrossRef](#)]
18. Medina-Quispe, F.; Castillo-Rojas, W.; Villegas, C. Metrics for the support of visual exploration of components in data mining models. *Ingeniare* **2020**, *28*, 596–611. [[CrossRef](#)]
19. Da Silva Lopes, M.A.; Dória Neto, A.D.; De Medeiros Martins, A. Parallel t-SNE Applied to Data Visualization in Smart Cities. *IEEE* **2020**, *8*, 11482–11490. [[CrossRef](#)]
20. Kopecká, M.; Hájek, M.; Jiménez-Alfaro, B.; Chytrý, M. The T-SNE Algorithm as a Tool to Improve the Quality of Reference Data Used in Accurate Mapping of Heterogeneous Non-forest Vegetation. *Remote Sens.* **2020**, *12*, 39. [[CrossRef](#)]
21. Liu, H.; Yang, J.; Ye, M.; James, S.C.; Tang, Z.; Dong, J.; Xing, T. Using T-distributed Stochastic Neighbor Embedding (t-SNE) for Cluster Analysis and Spatial Zone Delineation of Groundwater Geochemistry Data. *J. Hydrol.* **2021**, *597*, 126146. [[CrossRef](#)]
22. Figma Co. Available online: <https://www.figma.com/> (accessed on 20 April 2023).
23. Shore Labs. Kanban Tool. Available online: <https://kanbantool.com/es/metodologia-kanban> (accessed on 20 April 2023).
24. GitHub Inc. GitHub Platform. Available online: <https://github.com/> (accessed on 24 April 2023).
25. Sutherland, J. Future of scrum: Parallel pipelining of sprints in complex projects. In Proceedings of the Agile Development Conference (ADC'05), Denver, CO, USA, 24–29 July 2005; pp. 90–99.
26. Swimm Team: Popular Collaborative Coding Practices. Available online: <https://swimm.io/learn/code-collaboration/code-collaboration-styles-tools-and-best-practices/> (accessed on 2 July 2023).
27. Villamizar, M.; Garcés, O.; Castro, H.; Verano, M.; Salamanca, L.; Casallas, R.; Gil, S. Evaluating the Monolithic and the Microservice Architecture Pattern to Deploy Web Applications in the Cloud. In Proceedings of the 2015 10th Computing Colombian Conference (10CCC), Bogota, Colombia, 21–25 September 2015; pp. 583–590.
28. Jain, S.; Seeja, K.R.; Jindal, R. A fuzzy ontology framework in information retrieval using semantic query expansion. *Int. J. Inf. Manag. Data Insights* **2021**, *1*, 100009. [[CrossRef](#)]
29. Baeza-Yates, R.A.; Perleberg, C.H. Fast and Practical Approximate String Matching. *Inf. Process. Lett.* **1996**, *59*, 21–27. [[CrossRef](#)]
30. Yujian, L.; Bo, L. A normalised Levenshtein distance metric. *IEEE TPAMI* **2007**, *29*, 1091–1095. [[CrossRef](#)]
31. Schmidt, J. Usage of Visualisation Techniques in Data Science Workflows. *VISIGRAPP* **2020**, *3*, 309–316. [[CrossRef](#)]
32. Yalim, C.; Handley Holy, A.H. The effectiveness of visualisation techniques for supporting decision-making. In Proceedings of the Modeling, Simulation and Visualization Student Capstone Conference 2023, Suffolk, VA, USA, 20 April 2023. [[CrossRef](#)]
33. Bisong, E. Matplotlib and Seaborn. In *Building Machine Learning and Deep Learning Models on Google Cloud Platform*; Apress: Berkeley, CA, USA, 2019; pp. 151–165. [[CrossRef](#)]
34. Kurita, T. Principal Component Analysis (PCA). In *Computer Vision*; Springer: Cham, Switzerland, 2020. [[CrossRef](#)]
35. van der Maaten, L.J.P.; Hinton, G.E. Visualizing Data using t-SNE. *JMRL* **2008**, *9*, 2579–2605.
36. Hildenbrand, T.; Meyer, J. Intertwining lean and design thinking: Software product development from empathy to shipment. In *Software for People, Management for Professional*; Maedche, A., Botzenhardt, A., Neer, L., Eds.; Springer: Berlin/Heidelberg, Germany, 2012; pp. 217–237.
37. Srivastava, A.; Bhardwaj, S.; Saraswat, S. SCRUM model for agile methodology. In Proceedings of the 2017 International Conference on Computing, Communication and Automation (ICCCA), Greater Noida, India, 5–6 May 2017; pp. 864–869.
38. Sandhu, R.S. Role-Based Access Control. In *Advances in Computers*; Zelkowitz, M.V., Ed.; Elsevier: Amsterdam, The Netherlands, 1998; Volume 46, pp. 237–286. ISBN 978-0-12-012146-5.

39. ITU-T X.667; Information Technology—Procedures for the Operation of Object Identifier Registration Authorities: Generation of Universally Unique Identifiers and Their Use in Object Identifiers. ITU-T X-Series Recommendations; Telecommunication Standardization Sector of ITU (ITU-T): Geneva, Switzerland, 2012.
40. Leach, P.J.; Salz, R.; Mealling, M.H. RFC 4122—A Universally Unique Identifier (UUID) URN Namespace. Internet Engineering Task Force. 2005. Available online: <https://www.ietf.org/> (accessed on 10 November 2023).
41. Moriarty, K.; Kaliski, B.; Rusch, A. PKCS #5: Password-Based Cryptography Specification Version 2.1; IETF: 2017. Available online: <https://www.ietf.org/> (accessed on 10 November 2023).
42. Fisher, R.A. The use of multiple measurements in taxonomic problems. *Ann. Eugen.* **1936**, *7*, 179–188. [CrossRef]
43. Dirección General de Información en Salud. Datos Abiertos. Available online: [http://www.dgis.salud.gob.mx/contenidos/basesdedatos/Datos\\_Abiertos\\_gobmx.html](http://www.dgis.salud.gob.mx/contenidos/basesdedatos/Datos_Abiertos_gobmx.html) (accessed on 16 August 2023).
44. De Paula, D.; Cormican, K.; Dobrigkeit, F. From Acquaintances to Partners in Innovation: An Analysis of 20 Years of Design Thinking's Contribution to New Product Development. *IEEE Trans. Eng. Manag.* **2022**, *69*, 1664–1677. [CrossRef]

**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.