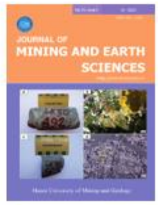
Journal of Mining and Earth Sciences

Website: http://jmes.humg.edu.vn



Researching and applying open-source WebGIS technology to build national-standard vector geographic database management software

Yen Quoc Phan*

Le Quy Don Technical University, Hanoi, Vietnam

ARTICLE INFO

Article history:
Received 01st Aug. 2023
Revised 16th Nov. 2023
Accepted 26th Nov. 2023

Keywords:
Geo database management system,
National standards,
Open Source Geospatial software,
Spatial Data Infrastructures (SDIs).

ABSTRACT

Geographical database management is a crucial component of spatial data infrastructures (SDIs) and is used by government organizations in provinces, regions, and countries. It plays an important role, especially during the 4.0 technology revolution and digital transformation to build e-government and digital government. A complete and scientific management of geographical databases is the most crucial core foundation of a system. The article applies open-source WebGIS technologies such as Geoserver map server, PostGIS data management software, Python programming language, Vue, and Openlayers map library to build spatial data management software. Spatial data is mainly in vector form, built according to national geographic database standards (QCVN 42: 2020/BTNMT). The system has main functional groups such as user account authorization and user groups; and geographic database management, including seven groups: borders, geodesy, population, topography, traffic, surface cover, and hydrology. In addition, the administration of some thematic classes such as drafting plans, managing equipment and units, etc., are performed. The software system is built in a service-oriented manner, serving as the core foundation to facilitate the development of many geo-portal applications.

*Corresponding author
E - mail: phanquocyen@lqdtu.edu.vn
DOI: 10.46326/JMES.2023.64(6).02

1. Introduction

Over the past decade, government organizations at the provincial, regional, national, and international levels, as well as NGOs (non-governmental organizations) and large companies, have been researching and developing a database with a geographical background (geodatabase) to increase access to spatial data for a wide range of users (Rajabifard & Williamson, 2001; Crompvoets et al., 2004). GIS data is stored on a server in another place called a cloud or data island. Users can access data mining anywhere, thereby saving resources and time and reducing data collection and maintenance work (Rajabifard & Williamson, 2001). The SDI Spatial Data Infrastructure is a series of coordination agreements on technology standards, institutional arrangements, and policies. It allows users to explore and use geospatial information for different purposes (Steiniger and Hunter, 2011). SDI is not just a technical basis but a complete framework covering political, technical, business, and social issues, consisting of 5 components: (i) spatial data (geographic database), (ii) technology (hardware and software), (iii) law and policy, (iv) people (data providers, service providers, and users), and (v) standards for data collection, representation, and transformation (Steiniger and Hunter, 2011; Kuhn, 2005).

SDI allows users to discover, analyse, update, and distribute spatial data from a data warehouse, ideally through one or more Web GIS services. Thus, the basic components of software include (i) client application software that can display, query, and analyse spatial data; (ii) a web service subscription server; (iii) a spatial data service server that enables the provision of data over the Internet and/or processing services such as data transformation and reference systems; (iv) spatial data warehousing; and (v) the computer GIS Desktop software that allows for the creation and maintenance of data (Figure 1) (Steiniger and Hunter, 2011).

The Global Geospatial Data Infrastructure Consortium (GSDI) was established in 1996, with 50 participating countries, to create an international forum for the exchange of NSDI ideas and to encourage international cooperation in the field. Since then, most countries and international

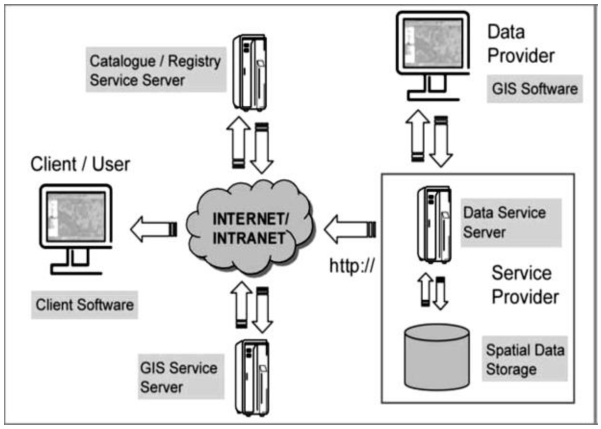


Figure 1. Software components required for SDI (Steiniger and Hunter, 2011).

organizations have focused their efforts on designing NSDI development projects in individual countries. Up to now, many countries around the world have enacted NSDI laws, such as the US, Germany, Poland, Bulgaria, Norway, Iceland, Macedonia, Estonia, Japan, Korea, Malaysia, Indonesia, Pakistan, and India (Tran, 2018).

Since GIS technology has been widely applied in Vietnam, there have been studies related to geospatial data management in web-based information systems, typically problems arising and standardized solutions, and online mapping (Tran, 2022). However, these studies are still moving towards thematic geographic data-sharing technology that does not comply with current geodatabase standards. Regarding Vietnam's SDI, the standards on collecting, representing, transforming, and sharing spatial data have been regulated in many documents (Ministry of Natural Resources and Environment, 2020; 2022a; 2022b), especially regulation 42/2020/BTNMT, which is a general standard and derivative of many other documents related to national technical regulations on basic geographic information standards. Many provinces and cities are also basing their efforts on these standards to build a geographic database, proceed to build a geographic portal and share the province's geospatial resources.

To build responsive software for geodatabase administration, there are many types of software products circulating in the world, especially software that distributes and stores spatial data over the Internet. In particular, the open-source geospatial software group (FOSS/FOSS4G) (Moreno-Sanchez, 2012) has become the foundation of the geospatial software ecosystem, providing core functions and services. It is considered to be on par with commercial software (Can et al., 2021). FOSS4G is important to developing countries because it allows them to develop their technology instead of having to import it. It provides the closest path to digitalization and is the basis for SDI creation in the context of scarce resources for system development and maintenance (Moreno-Sanchez, 2012; Molina and Bayarri, 2011).

Table 1. Summary of today's popular geospatial open source software packages/ libraries/ languages (OSGeoLive, 2019; Coetzee et al., 2020; Li et al., 2017).

No	Group	Software/ libraries/ languages
1	Front end	OpenLayers, Leaflet, Cesium, Geomajas, Mapbender, GeoExt, GeoMoose, GeoNode, Vue, Angular, React, etc.
2	Web GIS Server	GeoServer, MapServer, MapCache, Deegree, ncWMS, EOxServer, GeoNetwork, pycsw, PyWPS, MapProxy, QGIS Server, istSOS, 52 North SOS, 52 North WPS, Zoo Project, t-rex, Actinia, etc.
3	Spatial analysis library/ language Back end	GRASS GIS, GeoDjango, GDAL/OGR, GeoTools, GEOS, libLAS, JTS, PROJ4, CyberGIS Toolkit; GMT, Orfeo ToolBox, Mapnik, MapSlicer, R; PHP, Java, Python, NodeJS,
4	Database management system	PostGIS, SpatiaLite, Rasdaman, pgRouting, MySQL Spatial, MongoDB, CouchDB, Rasdaman

Table 1 shows that there are many technology options to build a WebGIS system. The technologies are highlighted by modern open-source tools, which are convenient and flexible to develop and adapt (Brovelli et al., 2017). Therefore, the purpose of this study is to apply those technologies to build a system of management and sharing of geographic background data according to standard 42/2020/BTNMT of the Ministry of Natural Resources and Environment. The system is built based on the MVC (MVT) model of the GeoDjango Framework, easy to inherit, adjust, and expand to many other applications that completely use only one geodatabase platform.

2. Methodology

2.1. Research process

System development, research, and selection of waterfall model methods to build and develop software with phases as shown in Figure 2 below:

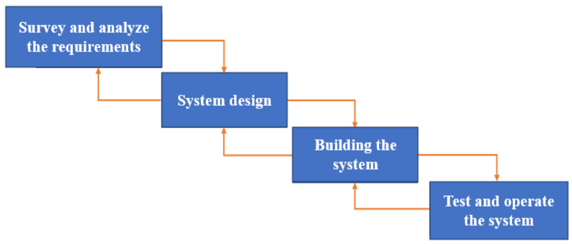


Figure 2. Software development phases.

Step 1, survey and analyze the requirements. Survey and evaluate the current status of the process, organize management and operation, then analyse the requirements for the system and make an implementation plan;

Step 2, system design: To describe the system, draw diagrams and specify the data processing flow between use-case actors affecting the system in an object-oriented manner, expressed in UML language. This is an important step in designing and developing the architecture of the whole system. Based on requirements analysis, design the database framework, create sample forms, collect, build, and standardize GIS data;

Step 3, building the system: Using WebGIS programming methods using backend and frontend technologies to build the software's functions;

Finally, test and operate the system. Use the Docker virtual host to install and package the environment to transfer to the server to operate the system. Access the software and use the functions through the installed Web link.

Designing a Web GIS system for geo-database management must achieve three objectives: first, simplifying the data sharing process; second, visualizing their interaction; and third, making it easy to inherit, develop, and extend the application.

To achieve those standards, the system's architecture must be designed according to the MVC (Model-Controller-View) model (Figure 3). This model architecture consists of three layers: the logical and data access layers, and the user application layer. These tiers are developed and built through Django's Framework (Django, 2022). Django follows the MVC design pattern. Still, due to Django's design, the Web Framework takes on the role of a controller and leaves the Model, View, and Templates blank for developers to fill in their code, so Django called the MTV model (Qu, 2017).

Model layer: The data access layer contains data-related procedures such as object definition, access, how to validate it, object relation mapping, etc.

Template layer: The presentation layer is responsible for providing an HTML page for the developer to fill in the resulting data, display it, and send it back to the requesting user.

View: The business logic layer, containing model access logic and definitions for the appropriate template, is the bridge between model and template.

2.2. Back-end technology solutions

Programming languages and libraries: The back-end uses the Python programming language, a high-level programming language. In addition, to achieve our goal of processing geo-background data, we integrated more software packages into the Django Framework, namely: (1) Django psycopg2 is a PostgreSQL and PostGIS adapter for Python; (2) Geo Django is a module for Django that turns it into a web framework for geospatial data. GeoDjango has the function to create WebGIS applications; its features include extensions to Django's ORMs for querying and manipulating spatial data; high-level Python interfaces for GIS geometry operations and data manipulation in various formats; (3) Django REST Framework is a powerful and flexible toolkit for building web APIs based on REST architecture. The Django REST Framework provides powerful model serialization, displaying data using standard function-based views or with views for more complex functions; (4) Django REST-framework-GIS: provides Django REST Framework geo plugins such as GeometryField fields and GeoJSON serializers; (5) Django GeoJSON is a toolkit for manipulating GeoJSON with Django for GeoDjango objects, query sets, lists, etc.; base view to serve GeoJSON map layers from models; GeoJSON models and form fields to avoid spatial databases at the server; (6) Django Redis Cache is server cache software for the Django Web Framework, featuring a key-value data structure (Django, 2022).

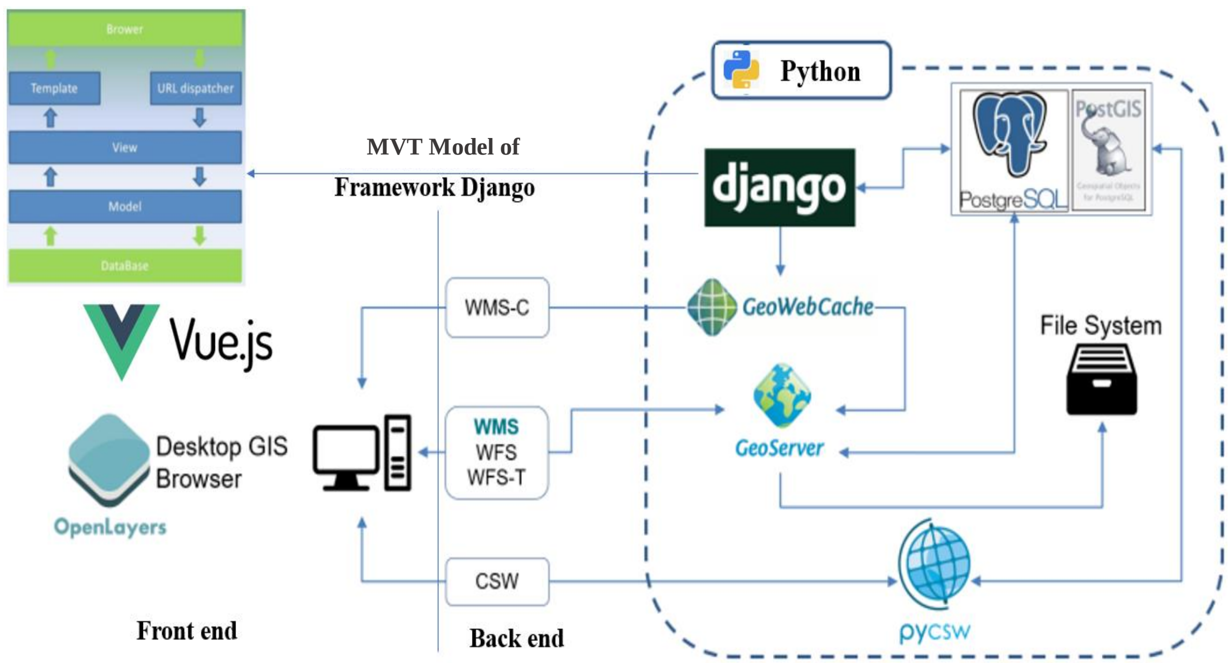


Figure 3. Architecture of the geographic database management software system.

Django distinguishes between an application and a project. An application is a system made for some function. A project is the host of many applications and their settings that form an entire website. To make an application reusable and a project extensible, a project can host multiple applications and an application can be migrated to multiple projects without problems. As shown in Table 2, the following are some basic commands to perform project and application creation in the software system.

Table 2. Basic commands to build a Django application (Crompvoets et al., 2004).

No	Commands	Command Description
1	django-admin startproject CSDL_SDI	Create a project
2	python manage.py startapp GOIGIAOTHONG	Create a Goigiathong application in a project
3	python manage.py runserver	Start the test server
4	python manage.py runserver 0.0.0.0:8000	Start the test server to access the internet

Map Server: The system's GIS server uses GeoServer, and the Web server uses Tomcat. GeoServer is a more mature open-source map service publishing platform that can easily publish WCS, WFS, WMS, and other map services. The data has been imported into the PostgreSQL database through PostGIS earlier. The required layers in the database are published, and after, the map service can be invoked in the front-end framework Vue by using OpenLayers with the space name and namespace URL and can be displayed visually in the front-end (Cui et al., 2022; Simoes and Cerciello, 2022).

Database management system: PostgreSQL was selected as the server-side database management software. So the database configuration in settings.py is shown below:

```
DATABASES = {
"default": {
"ENGINE": "django.db.backends.postgresql",
"NAME": "postgis_in_action",
"USER": "postgres",
"PASSWORD": "admin",
"HOST": "127.0.0.1",
"PORT": "5432" }
```

2.3. Front-end development environment

The technology that uses the client side consists of two basic libraries: Openlayer and Vue. Vue is a framework for building user interfaces. Unlike monolithic frameworks, Vue was designed from the ground up to enable and encourage step-by-step application development. To use Vue, the user must embed the library in the program. The embedding statement is as follows (Vue, 2022):

```
<script
src="https://cdn.jsdelivr.net/npm/vue/dist/vue.j
s"> </script>
```

OpenLayers is an open-source client-side library for creating web-based spatial data object interactions and displaying maps in WMS/WFS format standards in most current Web browsers, regardless of the server side. Since it is a client-side library, it requires no software or server-side installation; just execute the following statement to put the library to use (Openlayer, 2022):

```
import {Map, View} from 'ol'
```

How the Web retrieves map data from the Geoserver through the following code:

```
var congtrinhquansu = new VectorLayer({
title: ‘Cong trinh quan su’
source: new VectorSource({
format: new GeoJSON(),
loader: function(extent) {
$.ajax(‘http://localhost:8080/geoserver/wf’,
{ type: ‘GET’,
data: {
service: ‘WFS’,
version: ‘1.1.0’,
request: ‘GetFeature’,
typename: ‘Congtrinhquansu,
outputFormat: ‘application/json’, },
}).done(loadGeoFeatures);
}}})});
```

After the data is uploaded from the server in GeoJSON format, it can be displayed and presented as a map through the OpenLayers library.

3. Results and Discussion

3.1. Research data and area

The data and research area is the Geodatabase Vector dataset located in Gia Lai province. This regional data has been normalized for geographic background data according to standard 42/2020/BTNMT. The technical requirements to ensure the structural model and the content of the national geographic database at a scale of 1:100,000 include requirements for the structure and content model, requests for detailed data acquisition for the designed content and structure model, requirements for the quality of data captured in the structural model, and a graphical representation of the data. The data consists of seven packets, as shown in Figure 4.

The system operation data set ensures the geographical background database of a province according to regulations. Comply with geodatabase, dataset, and feature class regulations, especially the rules about feature class names, field names, and field domain scope. The identification code and the domain of the geographical feature are typed according to the Digital Regulation QCVN 71:2022/BTNMT on the structural model and the content of the national geographic database at a scale of 1:100.000 (Table 3).

Table 3. Description of the administrative territory on the mainland according to 42/2020/BTNMT.

No	MaNhanDang	PhienBan	NgayPhienBan	MaDonViHanhChinh
1	100NAD0100000001	1	23/03/2022	624
2	100NAD0100000002	1	23/03/2022	637
3	100NAD0100000003	1	23/03/2022	639
4	100NAD0100000004	1	23/03/2022	638
5	100NAD0100000005	1	23/03/2022	635
6	100NAD0100000006	1	23/03/2022	633
7	100NAD0100000007	1	23/03/2022	632
8	100NAD0100000008	1	23/03/2022	631
9	100NAD0100000009	1	23/03/2022	630
10	100NAD0100000010	1	23/03/2022	634
11	100NAD0100000011	1	23/03/2022	622
12	100NAD0100000012	1	23/03/2022	628
13	100NAD0100000013	1	23/03/2022	623
14	100NAD0200000014	1	23/03/2022	64

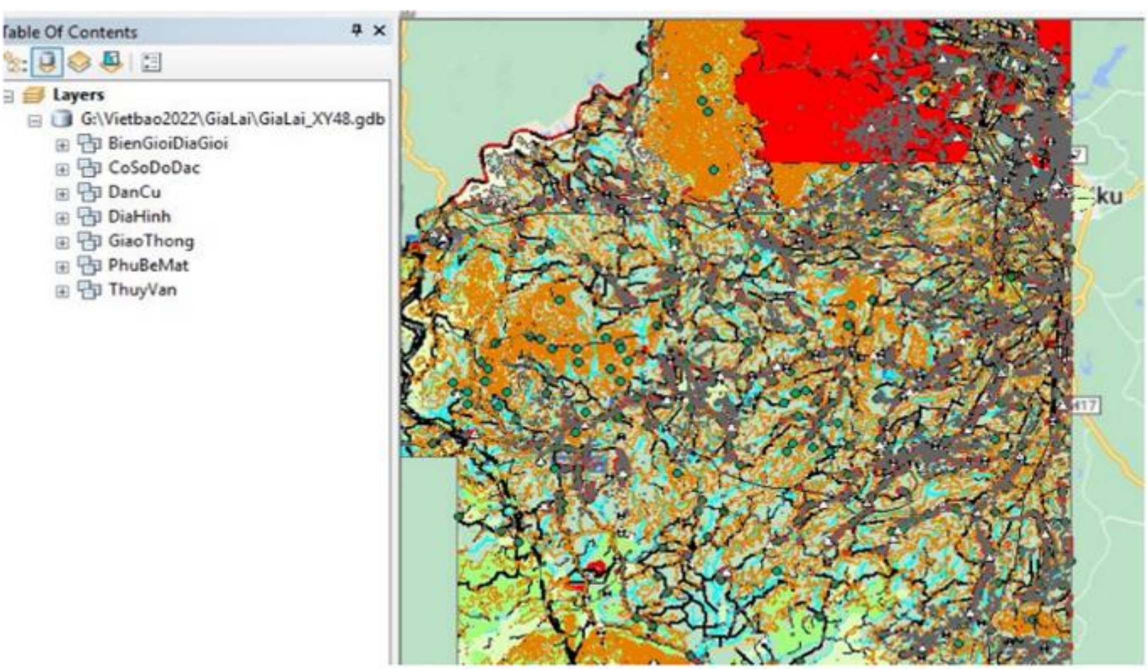


Figure 4. Research areas.

3.2. Result of geodata management software

3.2.1. Data model

In a system, database design is most important. It ensures the completeness and integrity of the data system. In addition to the tables of feature classes, which are designed according to standard 42/2020/BTNMT, the other data tables of the system also need to be fully designed to ensure logical management of content and application. In this software, the LopDuLieu board is used as the central board to configure the layer upload to the system. The information about images and videos describing the feature is integrated into a multimedia data table. In addition, to describe the geographic background data layers, there are metadata tables to describe the metadata of each feature class and style tables to store the display configuration of each layer. Each layer can be displayed in different ways. The data packet model of a system is shown in Figure 5 below:

3.2.2. Result of the main function of the system

The backend programming language is Python; the frontend is HTML and JavaScript; the libraries that use the server side are Django; and the client side is Openlayer and Vue. Implementing the system construction, some main results are summarized as follows:

- System login, logout, and authorization

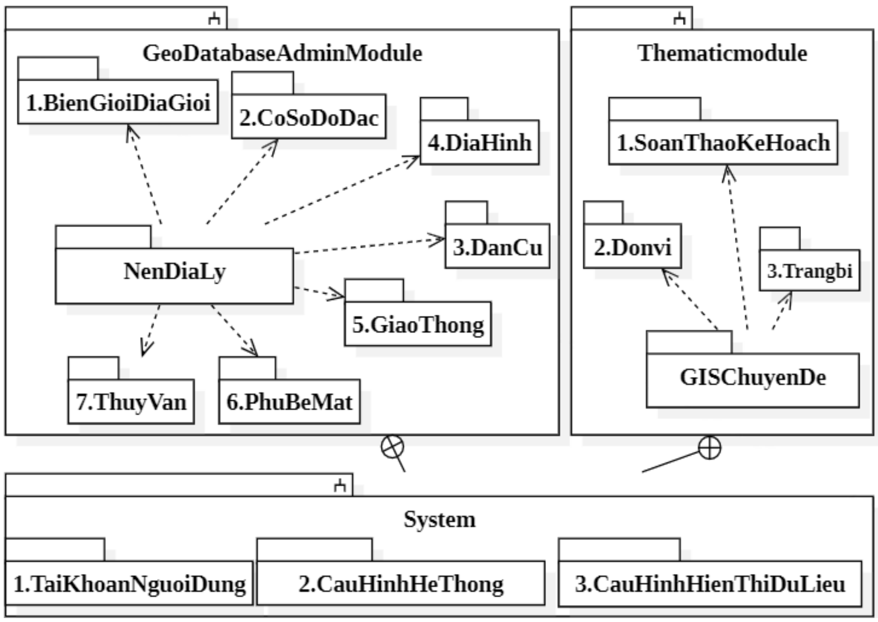


Figure 5. Data packet model of the system.

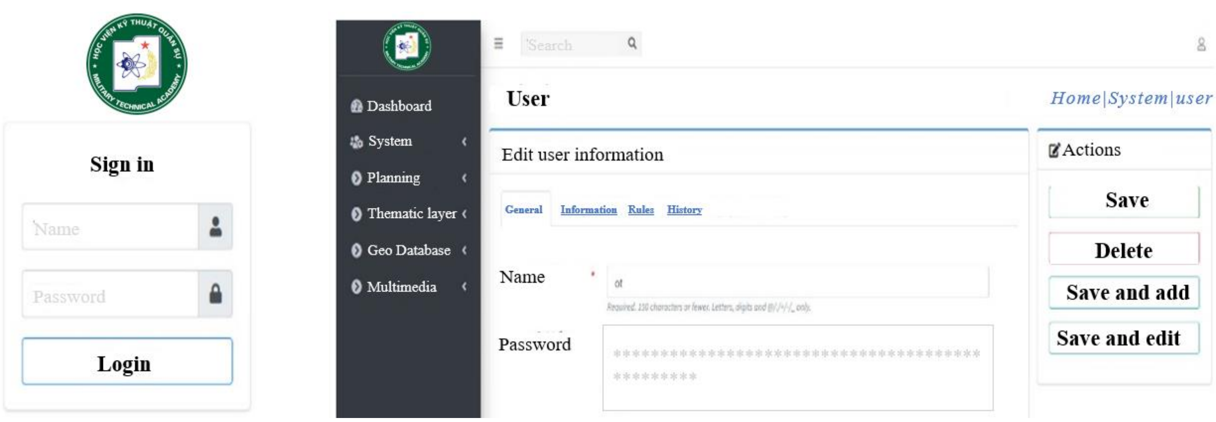


Figure 6. System login and authorization interface.

functions (Figure 6): The system is designed and built to decentralize users to each data table and each specific function. In addition to the system-wide administrative rights (Super admin), the system is also divided into rights groups such as data editing group, data query group, planning group, and decision support function use, etc.

- Geodatabase management function: This is the most complex function and takes up the most amount of work. In addition to the usual attribute data tables, the geodatabase has a total of more than 150 data tables in different formats, such as point, line, polygon, and raster. Raster data class for storing elevation digital model (DEM). All operations with the data layer are saved by the system and listed in the Recent Actions section. The main interface is shown in Figure 7 below.

- For each geo-background data layer, the layer can be selected by right-clicking in the black content area on the left side of the screen, shown in Figure 8, to select the data layer of the operating task layer and defense projects. The content on the right is the detailed viewing and manipulation area with the data table. For the administrative function, the administrator can search, view, add, edit, and delete information about the feature. Figure 9 below shows how to add a defense construction object to the data.

- For the system to be flexible in inheriting data, adjusting, and developing functions for new tasks, the software can add, edit, and delete information fields to any data table of the system without needing to act on the code (Figure 10).

- Application function to exploit the user's data: a multi-application-oriented system based on a centralized geographic data platform. Users can be divided into different groups (Figure 11): grouped by mining data, users who make plans, etc. To demonstrate the feasibility of customizing the user's application, the system implements several lookup and editing applications, such as the following interface:

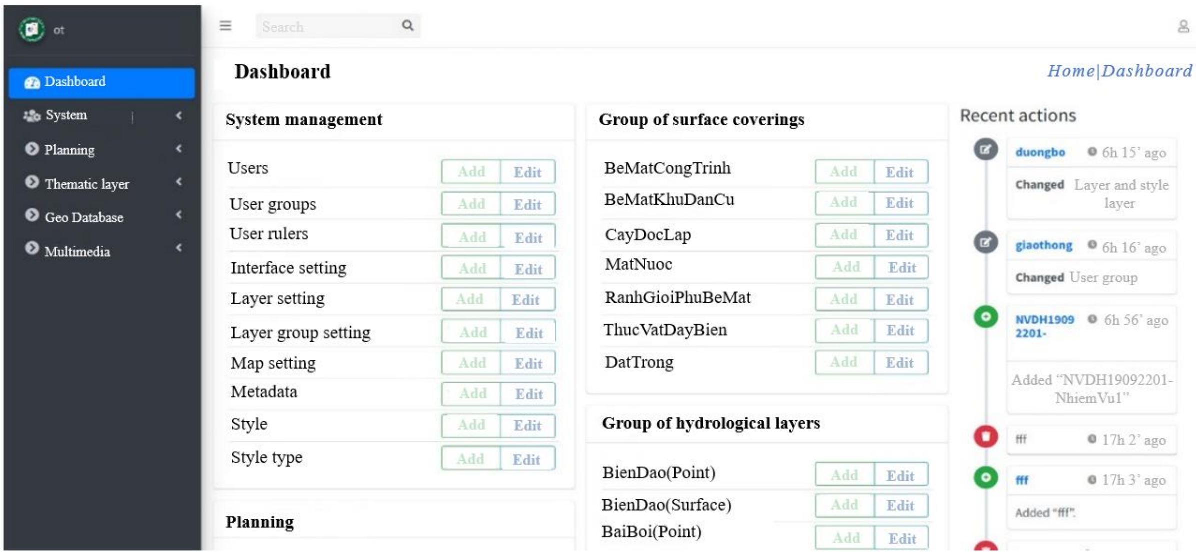


Figure 7. Dashboard interface of geo-based data management function.

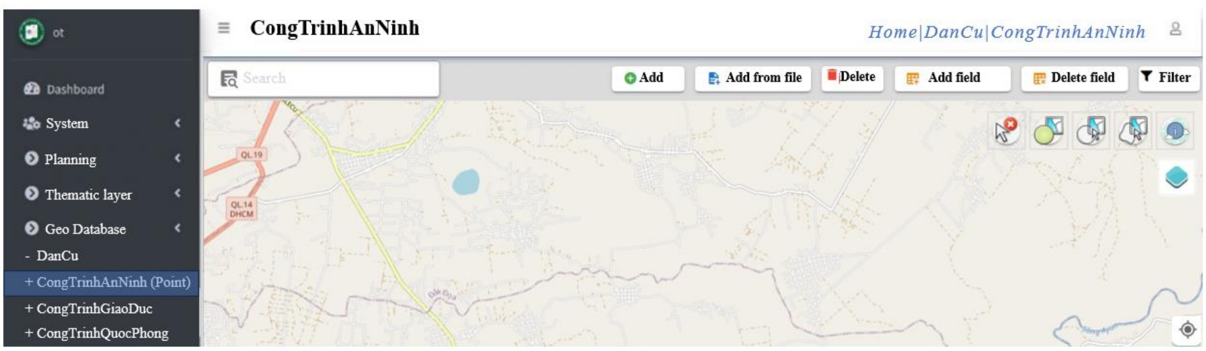


Figure 8. Administration interface of the security works.

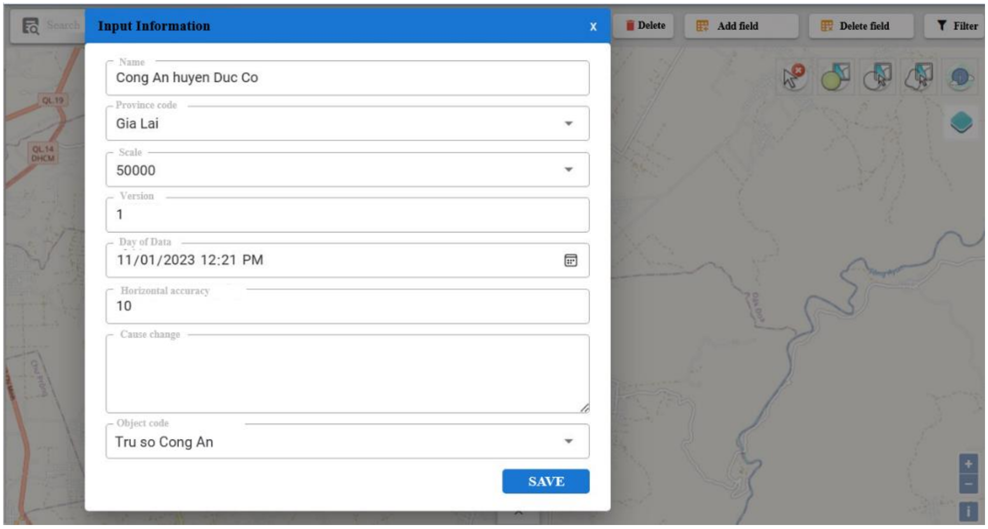


Figure 9. Adding data objects to the security building layer.

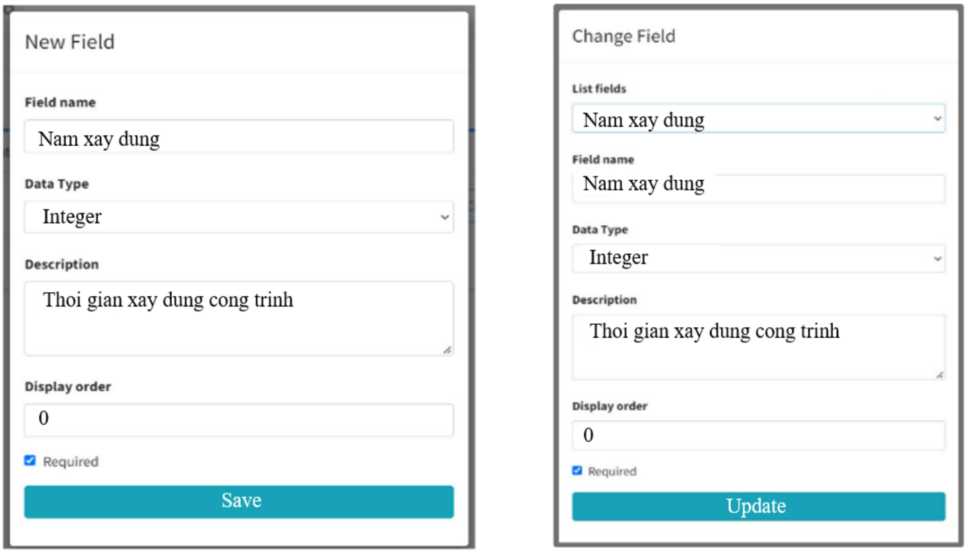


Figure 10. Functions for adding and changing field information.

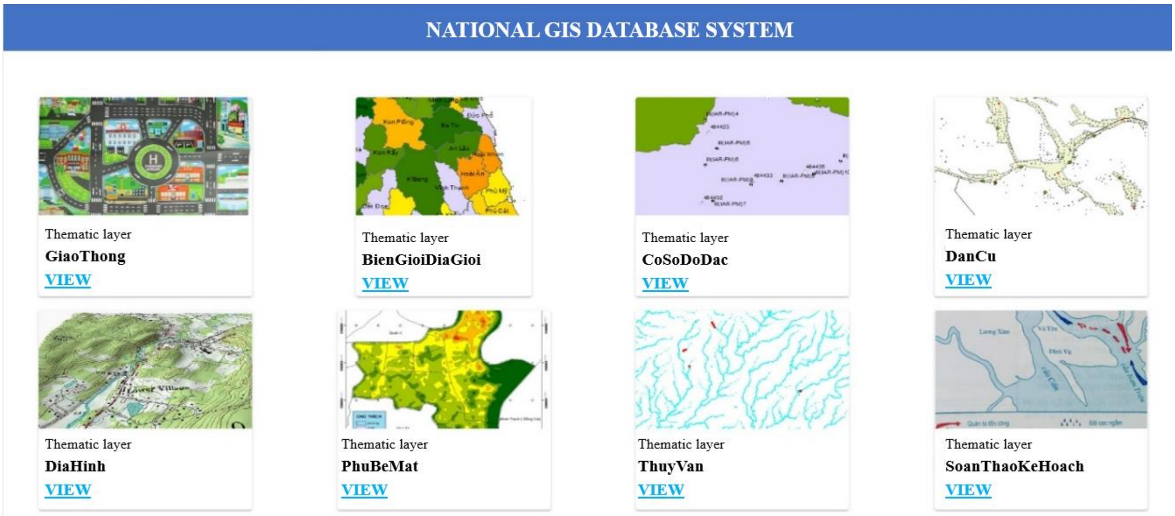


Figure 11. Selecting the field to perform the task of searching or drafting a plan.

After a successful login, users will be able to exploit documents and functions according to the system's authorization. The following (Figures 12 and 13) are two typical examples of data mining functions and document planning functions.

After clicking on the terrain layer discovery function, the software will download the terrain layer data configured by the administrator in the system. Figure 12 (left panel) shows a terrain group with five layers, of which two are displayed and three are not displayed. Here, users can toggle the display of data layers and search for and exploit information of interest. The right panel Figure 12 is a function of the portal to manage units and equipment based on national geographic data and allows users to search for units or equipment in any geographical area, serving to support search and rescue, defense, and security activities.

A more complex module demonstrating the superiority of centralized geographic data organization is the function of drafting plans or exchanging spatial information between different departments. According to the previous traditional method, only the TCP IP protocol was used to communicate and transmit information between different departments. Therefore, geographic background material often has to be copied to the machine parts. This is often inconvenient and takes a lot of time, memory, and power. With modern WebGIS technology, data is stored centrally on the server, using the HTTP protocol to share geographic background information between different departments. All departments drafted plans on the same map, the same area, and the same tasks, supporting the commander in effectively operating and monitoring the mission (Figure 13).

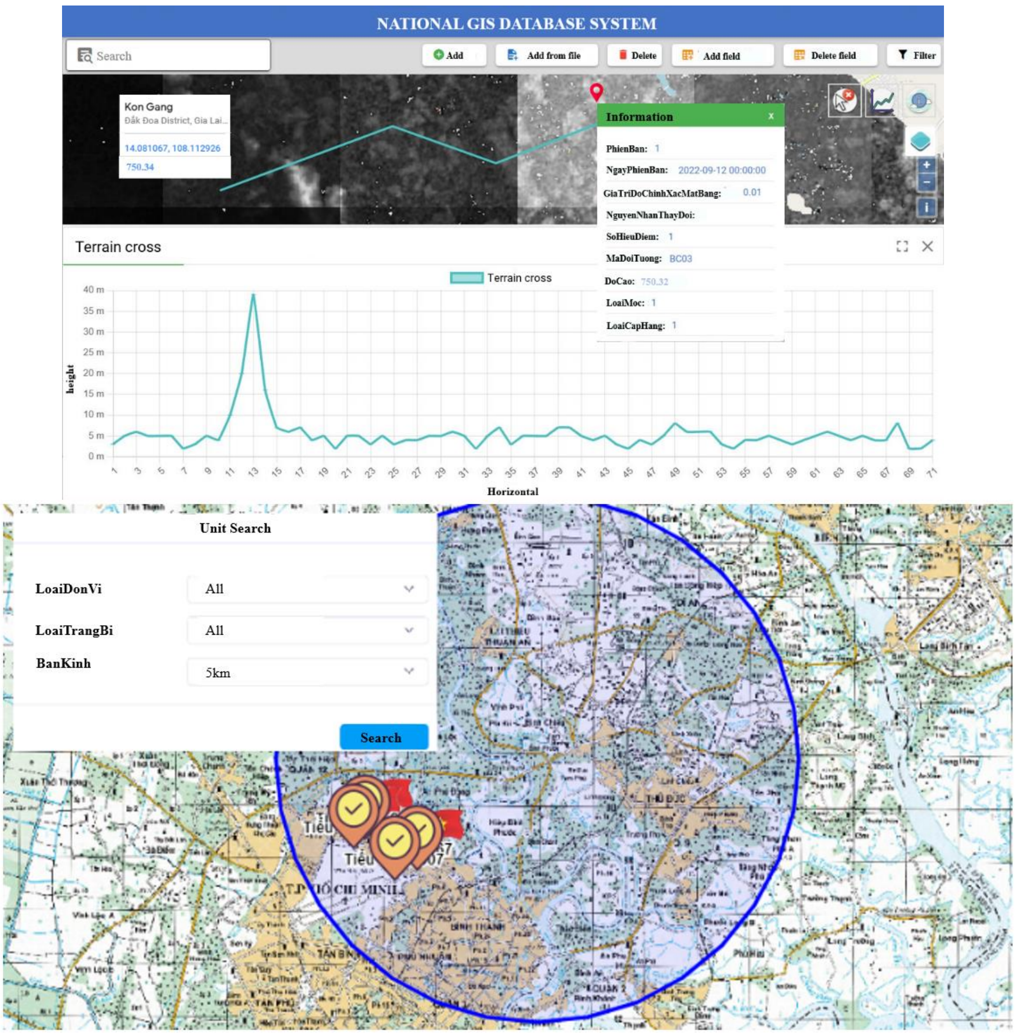


Figure 12. The portal supports searching for units by spatial radius.

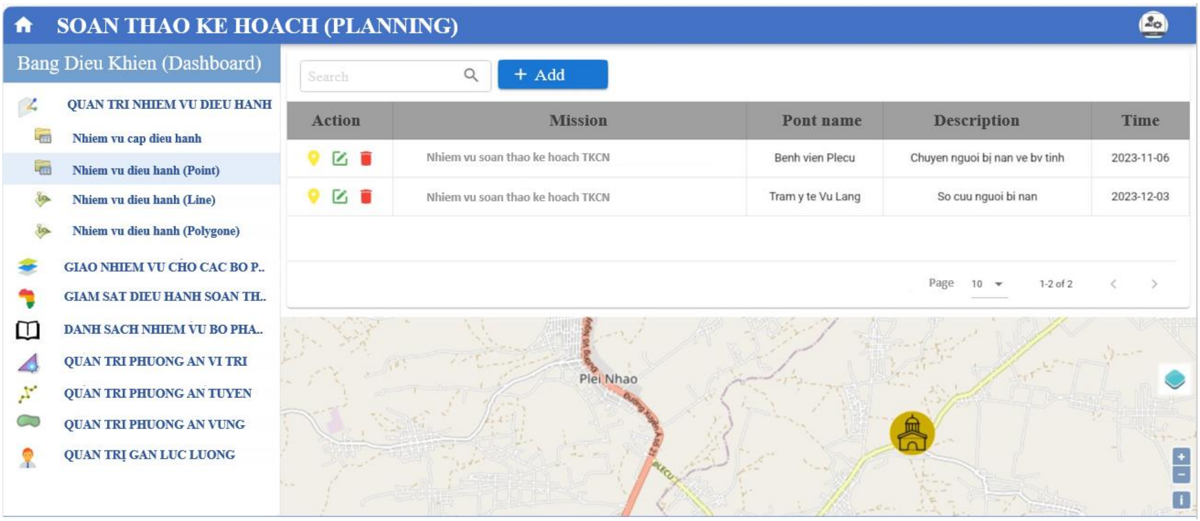


Figure 13. Information portal in support of planning and administering tasks.

4. Conclusion

Geodatabases and management, sharing, and exploitation software systems are among the most essential components of the national spatial infrastructure. Today, the development of geospatial technology, especially open-source geospatial software, is an important driving force in promoting SDI-based development.

The research has initially successfully tested popular open-source software in the geospatial industry to build a national geographic data management system according to the standards set by Vietnam. The system not only manages data according to standards but also designs and builds functions with great advantages, such as the ability for users to add new and modified data fields to the table and easily adjust and configure data for different user groups. This is a model similar to a geoportal.

Geospatial data has been big data for decades. New tools and technologies are now available to deal with big geodata analytics and visualization. Building a data management system according to national standards is an important first step to gradually managing big data for the system as a basis for building unified national geographic data. Especially with the explosion of technology, open-source technologies are increasingly developing and promoting the power of the community. The system is built as an OGC standard, and API interoperability allows for easy extensibility and interoperability with other clouds. The ease of porting applications or application components is key to big geodata governance. However, the research has only just begun, and this result is the foundation for the application of other open-source tools in big data management.

Contribution of authors

Yen Quoc Phan - conducted research for the entire article, including surveying and analysing the requirements, system design, building the system, testing and operating the system, and writing.

References

Brovelli, M. A., Minghini, M., Moreno-Sanchez, R., & Oliveira, R. (2017). Free and open source software for geospatial applications (FOSS4G) to support Future Earth. *International Journal of Digital Earth*, 10(4), 386-404.

Can, R. E. C. E. P., Kocaman, S. U. L. T. A. N., & Ok, A. O. (2021). A Webgis Framework for Semi-Automated Geodatabase Updating Assisted by Deep Learning. *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 43, 13-19.

Coetzee, S., Ivánová, I., Mitasova, H., & Brovelli, M. A. (2020). Open geospatial software and data: A review of the current state and a perspective into the future. *ISPRS International Journal of Geo-Information*, 9(2), 90.

Crompvoets, J., Bregt, A., Rajabifard, A., & Williamson, I. (2004). Assessing the worldwide developments of national spatial data clearinghouses. *International Journal of Geographical Information Science*, 18(7), 665-689.

Cui, L., Li, W., Wang, X., & Niu, X. (2022). Design and Implementation of Open-Source WebGIS System for Orchard Land Management. *Advances in Computer, Signals and Systems*, 6(3), 26-34.

Django, (2022). *Django documentation contents*. Website name. https://docs.djangoproject.com/en/4.2

Kuhn, W. (2005). *Introduction to Spatial Data Infrastructures.* Presentation held on March 14. Location. http://www.docstoc. com/docs/2697206/Introduction-to-Spatial--Data-Infrastructures.

Li, M., Minielly, C. M., Peak, D., Zeng, W., Lu, X., Lei, J., & Bell, S. (2017). An open source WebGIS framework for collaborative research in West Africa. *In Spatial Knowledge Initiative Conference, Banff, AB, Canada*, pp. 1-9.

Ministry of natural resources and environment, (2020). Technical regulations on structural model, national geographic database content, scale 1:10,000, 1:25,000 (15/2020/TT-BTNMT) (in Vietnamese).

Ministry of natural resources and environment, (2022a). National technical regulation on national topographic maps at scale 1:50,000, 1: 100,000 (QCVN 70: 2022/BTNMT) (in Vietnamese).

Ministry of natural resources and environment, (2022b). National technical regulation on national geographic database at scale 1:50,000, 1:100.000 (QCVN 71:2022/BTNMT) (in Vietnamese).

Molina, M., & Bayarri, S. (2011). A multinational SDI-based system to facilitate disaster risk management in the Andean Community. *Computers & Geosciences*, 37(9), 1501-1510.

Moreno-Sanchez, R. (2012). Free and Open Source Software for Geospatial Applications (FOSS4G): A mature alternative in the geospatial technologies arena. *Transactions in GIS*, 16(2), 81-88.

Openlayers, (2022). *Tutorials*, Website name. https://openlayers.org/doc/tutorials/

OSGeoLive, (2019). *Welcome to OSGeoLive*. Website name. https://live.osgeo.org/en/index.html

QCVN 42/2020/BTNMT, (2020). National technical regulation on basic geographic information standards (in Vietnamese).

Qu, B. (2017). Design and Implementation of a Geoscience & Technology Marketplace Web Portal. Master of Science. Texas A&M University.

Rajabifard, A., & Williamson, I. P. (2001). Spatial data infrastructures: concept, SDI hierarchy and future directions. *In Proceedings of GEOMATICS'80 Conference* (Vol. 10).

Simoes, J., & Cerciello, A. (2022). Serving Geospatial Data Using Modern and Legacy Standards: a Case Study from the Urban Health Domain. *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 48, 419-425.

Steiniger, S., & Hunter, A. J. (2011). Free and open source GIS software for building a spatial data infrastructure. Geospatial Free and Open Source Software in the 21st Century: Proceedings of the first Open Source Geospatial Research Symposium, OGRS 2009, Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 247-261.

Tran, B. G. (2018). Building, developing and managing Vietnam's national geospatial data infrastructure. *Proceedings of the National Conference on Surveying and Cartography,* Vietnam.

Tran, Q. V. (2022). Application of Web-GIS to construct an online map of Ecopark uburrn area, Van Giang District, Hung Yen Province. *Vietnam J. Agri. Sci.*, Vol. 20, No. 3: 325-332.

Vue. (2022), *Framework JavaScript*. https://vi.vuejs.org/v2/guide/comparison.html#ad